

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-6608

(43) 公開日 平成9年(1997)1月10日

(51) Int.Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 9/06	5 5 0		G 0 6 F 9/06	5 5 0 H
12/14	3 2 0		12/14	3 2 0 F

審査請求 未請求 請求項の数19 O L (全 26 頁)

(21) 出願番号 特願平7-156538

(22) 出願日 平成7年(1995)6月22日

(71) 出願人 000005821

松下電器産業株式会社

大阪府門真市大字門真1006番地

(72) 発明者 松崎 なつめ

大阪府門真市大字門真1006番地 松下電器
産業株式会社内

(72) 発明者 大森 基司

大阪府門真市大字門真1006番地 松下電器
産業株式会社内

(72) 発明者 館林 誠

大阪府門真市大字門真1006番地 松下電器
産業株式会社内

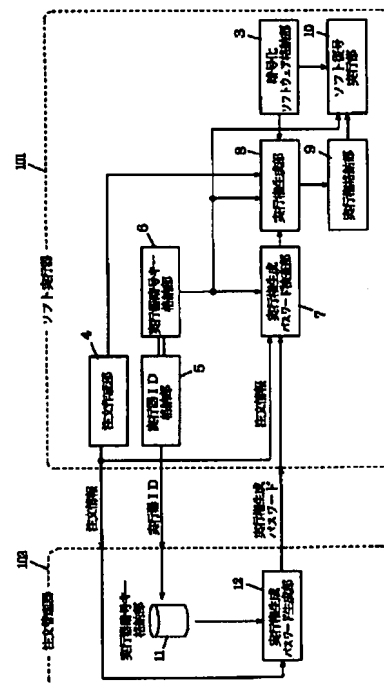
(74) 代理人 弁理士 小笠原 史朗

(54) 【発明の名称】 ソフトウェア保護システム

(57) 【要約】 (修正有)

【目的】 ソフトの実行権を通信路を用いて獲得する場合、通信路上のデータ、特にランダムな値の桁数を削減する。

【構成】 ソフト実行器101は、実行器IDおよび注文情報を生成し、これらを注文管理者102に送付することにより、ソフトウェアの注文を行なう。注文管理者102は、受け取った注文情報と、実行IDに対応する実行器秘密情報とに依存した情報である実行権生成パスワードを生成し、ソフト実行器101に送付する。ソフト実行器101は、この実行権生成パスワードが、注文管理者102に先に送付した注文情報や、自分自身の実行器秘密情報と整合している否かを判断し、整合している場合は、注文情報に対応したソフトウェアの実行権を生成し、当該ソフトウェアを実行する。



【特許請求の範囲】

【請求項1】 提供されたソフトウェアを実行する1つ以上のソフト実行器と、各ソフト実行器と通信経路を介して接続され、かつ各ソフト実行器から受けたソフトウェアの注文を管理する注文管理者とを備えたソフトウェア保護システムであって、

各前記ソフト実行器は、

固有の実行器IDを格納する実行器ID格納手段と、固有の実行器秘密情報を格納する第1の秘密情報格納手段と、

1つ以上のソフトウェアの実行権の注文情報を作成して蓄積保持する注文作成手段とを含み、

前記実行器IDおよび前記注文情報は、前記通信路を介して前記注文管理者に送付され、

前記注文管理者は、

各前記ソフト実行器に格納された全ての実行器秘密情報を格納する第2の秘密情報格納手段と、

前記ソフト実行器から受け取った実行器IDに対応する実行器秘密情報を前記第2の秘密情報格納手段から獲得し、この獲得した実行器秘密情報およびソフト実行器から受け取った注文情報に依存した実行権生成パスワードを生成する実行権生成パスワード生成手段とを含み、

前記実行権生成パスワードは、前記通信路を介して前記ソフト実行器に送付され、

各前記ソフト実行器は、さらに前記注文作成手段に蓄積保持された注文情報と、前記第1の秘密情報格納手段に格納された実行器秘密情報とを用いて、前記注文管理者から受け取った実行権生成パスワードの正当性を検査する実行権生成パスワード検査手段と、

前記実行権生成パスワード検査手段による検査の結果、前記実行権生成パスワードの正当性が確認されたときに、前記注文作成手段に蓄積保持された注文情報に基づいて、注文されたソフトウェアの実行権を生成する実行権生成手段と、

前記実行権が存在するソフトウェアを、当該実行権に示される実行条件に従って実行するソフト実行手段とを含む、ソフトウェア保護システム。

【請求項2】 前記実行権生成パスワード生成手段は、前記ソフト実行器から受け取った注文情報と、前記第2の秘密情報格納手段から獲得した実行器秘密情報とを、出力が入力ビットのすべてに関係するようなデータ圧縮関数に入力し、このデータ圧縮関数の出力結果を、前記実行権生成パスワードとして出力し、

前記実行権生成パスワード検査手段は、前記注文作成手段に蓄積保持された注文情報と、前記第1の秘密情報格納手段に格納された実行器秘密情報とを、前記実行権生成パスワード生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、その出力結果が注文管理者から受け取った実行権生成パスワードと一致したとき、当該実行権生成パスワードが正当であると判断すること

を特徴とする、請求項1に記載のソフトウェア保護システム。

【請求項3】 前記注文管理者は、公開鍵署名方式に用いられる秘密情報を格納する第3の秘密情報格納手段をさらに含み、

前記実行権生成パスワード生成手段は、前記第3の秘密情報格納手段に格納された秘密情報と、前記第2の秘密情報格納手段から獲得した実行器秘密情報と、前記ソフト実行器から受け取った注文情報とを用いて、公開鍵署名方式によってデジタル署名された実行権生成パスワードを生成し、

前記ソフト実行器は、前記公開鍵署名方式に用いられる秘密情報に対応した公開情報を格納する公開情報格納手段をさらに含み、

前記実行権生成パスワード検査手段は、前記公開情報格納手段に格納された公開情報と、前記注文作成手段に蓄積保持された注文情報と、前記第1の秘密情報格納手段に格納された実行器秘密情報とを用いて、前記公開鍵署名方式に対応した署名確認方式によって、前記注文管理者から受け取った実行権生成パスワードの正当性を検査することを特徴とする、請求項1に記載のソフトウェア保護システム。

【請求項4】 提供されたソフトウェアを実行する1つ以上のソフト実行器と、各ソフト実行器と通信経路を介して接続され、かつ各ソフト実行器から受けたソフトウェアの注文を管理する注文管理者とを備えたソフトウェア保護システムであって、

各前記ソフト実行器は、

固有の実行器IDを格納する実行器ID格納手段と、

固有の実行器秘密情報を格納する第1の秘密情報格納手段と、

1つ以上のソフトウェアの実行権の注文情報を作成して蓄積保持する注文作成手段と、

前記注文作成手段で注文情報が作成される毎に変化する変化値を生成して蓄積保持する変化値生成手段とを含み、

前記実行器ID、前記注文情報および前記変化値は、前記通信路を介して前記注文管理者に送付され、

前記注文管理者は、

各前記ソフト実行器に格納された全ての実行器秘密情報を格納する第2の秘密情報格納手段と、

前記ソフト実行器から受け取った実行器IDに対応する実行器秘密情報を前記第2の秘密情報格納手段から獲得し、この獲得した実行器秘密情報、ソフト実行器から受け取った注文情報および変化値に依存した実行権生成パスワードを生成する実行権生成パスワード生成手段とを含み、

前記実行権生成パスワードは、前記通信路を介して前記ソフト実行器に送付され、

各前記ソフト実行器は、さらに前記注文作成手段に蓄積

保持された注文情報と、前記変化値生成手段に蓄積保持された変化値と、前記第1の秘密情報格納手段に格納された実行器秘密情報とを用いて、前記注文管理者から受け取った実行権生成パスワードの正当性を検査する実行権生成パスワード検査手段と、

前記実行権生成パスワード検査手段による検査の結果、前記実行権生成パスワードの正当性が確認されたときに、前記注文作成手段に蓄積保持された注文情報に基づいて、注文されたソフトウェアの実行権を生成する実行権生成手段と、

前記実行権が存在するソフトウェアを、当該実行権に示される実行条件に従って実行するソフト実行手段とを含む、ソフトウェア保護システム。

【請求項5】 前記実行権生成パスワード生成手段は、前記ソフト実行器から受け取った注文情報および変化値と、前記第2の秘密情報格納手段から獲得した実行器秘密情報とを、出力が入力ビットのすべてに関係するようなデータ圧縮関数に入力し、このデータ圧縮関数の出力結果を、前記実行権生成パスワードとして出力し、前記実行権生成パスワード検査手段は、前記注文作成手段に蓄積保持された注文情報と、前記変化値生成手段に蓄積保持された変化値と、前記第1の秘密情報格納手段に格納された実行器秘密情報とを、前記実行権生成パスワード生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、その出力結果が注文管理者から受け取った実行権生成パスワードと一致したとき、当該実行権生成パスワードが正当であると判断することの特徴とする、請求項4に記載のソフトウェア保護システム。

【請求項6】 提供されたソフトウェアを実行する1つ以上のソフト実行器と、各ソフト実行器と通信経路を介して接続され、かつ各ソフト実行器から受けたソフトウェアの注文を管理する注文管理者とを備えたソフトウェア保護システムであって、

各前記ソフト実行器は、

固有の実行器IDを格納する実行器ID格納手段と、

固有の実行器秘密情報を格納する第1の秘密情報格納手段と、

1つ以上のソフトウェアの実行権の注文情報を作成して蓄積保持する注文作成手段と、

前記注文作成手段で注文情報が作成される毎にタイムスタンプを生成して蓄積保持する第1のタイムスタンプ生成手段とを含む、

前記実行器IDおよび前記注文情報は、前記通信路を介して前記注文管理者に送付され、

前記注文管理者は、

前記ソフト実行器から実行器IDおよび注文情報を受け取ると、タイムスタンプを生成する第2のタイムスタンプ生成手段と、

各前記ソフト実行器に格納された全ての実行器秘密情報を格納する第2の秘密情報格納手段と、

前記ソフト実行器から受け取った実行器IDに対応する実行器秘密情報を前記第2の秘密情報格納手段から獲得し、この獲得した実行器秘密情報と、ソフト実行器から受け取った注文情報と、前記第2のタイムスタンプ生成手段で生成されたタイムスタンプとに依存した実行権生成パスワードを生成する実行権生成パスワード生成手段とを含む、

前記実行権生成パスワードは、前記通信路を介して前記ソフト実行器に送付され、

各前記ソフト実行器は、さらに前記注文作成手段に蓄積保持された注文情報と、前記第1のタイムスタンプ生成手段に蓄積保持されたタイムスタンプと、前記第1の秘密情報格納手段に格納された実行器秘密情報とを用いて、前記注文管理者から受け取った実行権生成パスワードの正当性を検査する実行権生成パスワード検査手段と、

前記実行権生成パスワード検査手段による検査の結果、前記実行権生成パスワードの正当性が確認されたときに、前記注文作成手段に蓄積保持された注文情報に基づいて、注文されたソフトウェアの実行権を生成する実行権生成手段と、

前記実行権が存在するソフトウェアを、当該実行権に示される実行条件に従って実行するソフト実行手段とを含む、ソフトウェア保護システム。

【請求項7】 提供されたソフトウェアを実行する1つ以上のソフト実行器と、各ソフト実行器と通信経路を介して接続され、かつ各ソフト実行器から受けたソフトウェアの注文を管理する注文管理者とを備えたソフトウェア保護システムであって、

各前記ソフト実行器は、

固有の実行器IDを格納する実行器ID格納手段と、

固有の実行器秘密情報を格納する第1の秘密情報格納手段と、

1つ以上のソフトウェアの実行権の注文情報を作成して蓄積保持する注文作成手段と、

前記注文情報と前記実行器秘密情報に依存した注文認証情報を生成して蓄積保持する注文認証情報生成手段とを含む、

前記実行器ID、前記注文情報および前記注文認証情報は、前記通信路を介して前記注文管理者に送付され、

前記注文管理者は、

各前記ソフト実行器に格納された全ての実行器秘密情報を格納する第2の秘密情報格納手段と、

前記ソフト実行器から受け取った実行器IDに対応する実行器秘密情報を前記第2の秘密情報格納手段から獲得し、この獲得した実行器秘密情報、ソフト実行器から受け取った注文情報および注文認証情報を用いて、ソフト実行器および注文情報の認証を行い、この認証結果が正当である場合に、当該注文認証情報を注文を識別する注文識別情報として扱う実行器認証手段と、

前記実行器認証手段における認証結果が正当である場合にのみ、前記注文情報および前記注文識別情報に依存した実行権生成パスワードを生成する実行権生成パスワード生成手段とを含み、

前記実行権生成パスワードは、前記注文識別情報と共に、前記通信路を介して前記ソフト実行器に送付され、各前記ソフト実行器は、さらに前記注文作成手段に蓄積保持された注文情報と、前記注文認証情報生成手段に蓄積保持された注文認証情報とを用いて、前記注文管理者から受け取った実行権生成パスワードの正当性を検査する実行権生成パスワード検査手段と、

前記実行権生成パスワード検査手段による検査の結果、前記実行権生成パスワードの正当性が確認されたときに、前記注文作成手段に蓄積保持された注文情報に基づいて、注文されたソフトウェアの実行権を生成する実行権生成手段と、

前記実行権が存在するソフトウェアを、当該実行権に示される実行条件に従って実行するソフト実行手段とを含む、ソフトウェア保護システム。

【請求項8】 前記注文認証情報生成手段は、前記注文作成手段によって作成された注文情報と、前記第1の秘密情報格納手段に格納された実行器秘密情報とを、出力が入力ビットのすべてに関係するようなデータ圧縮関数に入力し、このデータ圧縮関数の出力結果を、前記注文認証情報として出力し、

前記実行器認証手段は、前記第2の秘密情報格納手段から獲得した実行器秘密情報と、ソフト実行器から受け取った注文情報とを、前記注文認証情報生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、このデータ圧縮関数の出力結果がソフト実行器から受け取った注文認証情報と一致する場合に、注文を行ったソフト実行器および注文情報が正当であることを認証し、前記実行権生成パスワード生成手段は、前記ソフト実行器から受け取った注文情報および前記注文識別情報を、前記注文認証情報生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、このデータ圧縮関数の出力結果を、前記実行権生成パスワードとして出力し、前記実行権生成パスワード検査手段は、前記注文作成手段に蓄積保持された注文情報と、前記注文認証情報生成手段に蓄積保持された注文認証情報とを、前記注文認証情報生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、その出力結果が注文管理者から受け取った実行権生成パスワードと一致したとき、当該実行権生成パスワードが正当であると判断することの特徴とする、請求項7に記載のソフトウェア保護システム。

【請求項9】 提供されたソフトウェアを実行する1つ以上のソフト実行器と、各ソフト実行器と通信経路を介して接続され、かつ各ソフト実行器から受けたソフトウェアの注文を管理する注文管理者とを備えたソフトウェア保護システムであって、

各前記ソフト実行器は、

固有の実行器IDを格納する実行器ID格納手段と、固有の実行器秘密情報を格納する第1の秘密情報格納手段と、

1つ以上のソフトウェアの実行権の注文情報を作成して蓄積保持する注文作成手段と、

前記注文作成手段で注文情報が作成される毎に変化し、所定のデータ構造または意味を有する変化値を生成して蓄積保持する変化値生成手段と、

前記注文情報および前記実行器秘密情報に依存した値を求め、この求めた値によって前記変化値を変換することにより、注文認証情報を生成して蓄積保持する注文認証情報生成手段とを含み、

前記実行器ID、前記注文情報および前記注文認証情報は、前記通信路を介して前記注文管理者に送付され、前記注文管理者は、

各前記ソフト実行器に格納された全ての実行器秘密情報を格納する第2の秘密情報格納手段と、

前記ソフト実行器から受け取った実行器IDに対応する実行器秘密情報を前記第2の秘密情報格納手段から獲得し、この獲得した実行器秘密情報とソフト実行器からの注文情報とに依存した値を求め、この値を用いてソフト実行器からの注文認証情報を逆変換する逆変換手段と、前記逆変換手段の出力が前記変化値と同じデータ構造または意味を有する場合に、ソフト実行器および注文情報が正当であることを認証する実行器認証手段と、

前記実行器認証手段における認証結果が正当である場合にのみ、前記注文情報に依存した実行権生成パスワードを生成する実行権生成パスワード生成手段とを含み、前記実行権生成パスワードは、前記通信路を介して前記ソフト実行器に送付され、

各前記ソフト実行器は、さらに前記注文作成手段に蓄積保持された注文情報と、前記注文認証情報生成手段に蓄積保持された注文認証情報とを用いて、前記注文管理者から受け取った実行権生成パスワードの正当性を検査する実行権生成パスワード検査手段と、

前記実行権生成パスワード検査手段による検査の結果、前記実行権生成パスワードの正当性が確認されたときに、前記注文作成手段に蓄積保持された注文情報に基づいて、注文されたソフトウェアの実行権を生成する実行権生成手段と、

前記実行権が存在するソフトウェアを、当該実行権に示される実行条件に従って実行するソフト実行手段とを含む、ソフトウェア保護システム。

【請求項10】 前記注文認証情報生成手段は、前記注文作成手段によって作成された注文情報と、前記第1の秘密情報格納手段に格納された実行器秘密情報とを、出力が各入力ビットのすべてに関係するようなデータ圧縮関数に入力し、その出力結果と前記変化値との排他的論理和を注文認証情報として出力し、前記逆変換手段は、

ソフト実行器からの注文情報と、前記第2の秘密情報格納手段から獲得した実行器秘密情報とを、前記注文認証情報生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、その出力結果とソフト実行器からの注文認証情報との排他的論理和を演算することにより、当該注文認証情報を逆変換し、

前記実行権生成パスワード生成手段は、前記ソフト実行器から受け取った注文情報および前記注文識別情報を、前記注文認証情報生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、このデータ圧縮関数の出力結果を、前記実行権生成パスワードとして出力し、前記実行権生成パスワード検査手段は、前記注文作成手段に蓄積保持された注文情報と、前記注文認証情報生成手段に蓄積保持された注文認証情報とを、前記注文認証情報生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、その出力結果が注文管理者から受け取った実行権生成パスワードと一致したとき、当該実行権生成パスワードが正当であると判断することを特徴とする、請求項9に記載のソフトウェア保護システム。

【請求項11】 前記実行権生成パスワード生成手段は、ソフト実行器から受け取った注文情報に加えて、前記第2の秘密情報格納手段から獲得した実行器秘密情報に依存した実行権生成パスワードを生成し、前記実行権生成パスワード検査手段は、前記注文作成手段に蓄積保持された注文情報と、前記第1の秘密情報格納手段に格納されている実行器秘密情報とを用いて、前記注文管理者から受け取った実行権生成パスワードの正当性を検査することを特徴とする、請求項7または9に記載のソフトウェア保護システム。

【請求項12】 前記注文管理者は、公開鍵署名方式に用いられる秘密情報を格納する第3の秘密情報格納手段をさらに含み、

前記実行権生成パスワード生成手段は、前記第3の秘密情報格納手段に格納された秘密情報と、前記第2の秘密情報格納手段から獲得した実行器秘密情報と、前記ソフト実行器から受け取った注文情報および注文認証情報とを用いて、公開鍵署名方式によってデジタル署名された実行権生成パスワードを生成し、

前記ソフト実行器は、前記公開鍵署名方式に用いられる秘密情報に対応した公開情報を格納する公開情報格納手段をさらに含み、

前記実行権生成パスワード検査手段は、前記公開情報格納手段に格納された公開情報と、前記注文作成手段に蓄積保持された注文情報と、前記注文認証情報生成手段に蓄積保持された注文認証情報と、前記第1の秘密情報格納手段に格納された実行器秘密情報とを用いて、前記公開鍵署名方式に対応した署名確認方式によって、前記注文管理者から受け取った実行権生成パスワードの正当性を検査することを特徴とする、請求項7または9に記載のソフトウェア保護システム。

【請求項13】 提供されたソフトウェアを実行する1つ以上のソフト実行器と、各ソフト実行器と通信経路を介して接続され、かつ各ソフト実行器から受けたソフトウェアの注文を管理する注文管理者とを備えたソフトウェア保護システムであって、

各前記ソフト実行器は、

固有の実行器IDを格納する実行器ID格納手段と、固有の実行器秘密情報を格納する第1の秘密情報格納手段と、

1つ以上のソフトウェアの実行権の注文情報を作成して蓄積保持する注文作成手段とを含み、

前記実行器IDおよび前記注文情報は、前記通信路を介して前記注文管理者に送付され、

前記注文管理者は、

第1の実行権生成モジュールを格納する第1のモジュール格納手段と、

各前記ソフト実行器に格納された全ての実行器秘密情報を格納する第2の秘密情報格納手段と、

前記ソフト実行器からの注文情報と、前記第2の秘密情報格納手段から獲得した実行器秘密情報とに依存した値を計算する第1の計算手段と、

前記第1の計算手段の計算結果を用いて、前記第1の実行権生成モジュールを変換することにより、実行権生成パスワードを生成する実行権生成パスワード生成手段とを含み、

前記実行権生成パスワードは、前記通信路を介して前記ソフト実行器に送付され、

各ソフト実行器は、さらに第2の実行権生成モジュールを格納する第2のモジュール格納手段と、

前記注文作成手段に蓄積保持された注文情報と、前記第1の秘密情報格納手段に格納された実行器秘密情報とに依存した値を計算する第2の計算手段と、

前記第2の計算手段の計算結果を用いて、前記注文管理者からの実行権生成パスワードを逆変換することにより、第1の実行権生成モジュールを生成する実行権生成パスワード逆変換手段と、

前記実行権生成パスワード逆変換手段により生成された第1の実行権生成モジュールと、前記第2のモジュール格納手段に格納されている第2の実行権生成モジュールとを用いて、前記注文作成手段に蓄積保持された注文情報に基づくソフトウェアの実行権を生成する実行権生成手段と、

実行権生成後、第一の実行権生成モジュールを消去するモジュール消去手段と、

前記実行権が存在するソフトウェアを、当該実行権に示される実行条件に従って実行するソフト実行手段とを含む、ソフトウェア保護システム。

【請求項14】 前記第1のモジュール格納手段に格納されている第1の実行権生成モジュールは、所定のデータ構造や意味をもっており、

前記実行権生成手段は、前記実行権生成パスワード逆変換手段の逆変換結果が前記第1の実行権生成モジュールと同じデータ構造や意味を持っている場合にのみ、第1の実行権生成モジュールと、第2の実行権生成モジュールとを用いて、前記注文作成手段に蓄積保持された注文情報に基づくソフトウェアの実行権を生成することを特徴とする、請求項13に記載のソフトウェア保護システム。

【請求項15】 前記注文管理者は、公開鍵署名方式に用いられる秘密情報を格納する第3の秘密情報格納手段をさらに含み、

前記実行権生成パスワード生成手段は、前記第1の計算手段の計算結果と、前記第3の秘密情報格納手段に格納された秘密情報とを用いて、前記第1の実行権生成モジュールを変換することにより、公開鍵署名方式によってデジタル署名された実行権生成パスワードを生成し、前記ソフト実行器は、前記公開鍵署名方式に用いられる秘密情報に対応した公開情報を格納する公開情報格納手段をさらに含み、

前記実行権生成パスワード逆変換手段は、前記第2の計算手段の計算結果と、前記公開情報とを用い、前記公開鍵署名方式に対応した署名確認方式によって、前記注文管理者からの実行権生成パスワードを逆変換することを特徴とする、請求項13に記載のソフトウェア保護システム。

【請求項16】 前記ソフト実行器に提供されるソフトウェアは、ソフト固有情報を含んで暗号化されており、前記実行権生成手段は、前記実行権生成パスワード検査手段が前記注文管理者からの実行権生成パスワードの正当性を確認したとき、前記注文作成手段に蓄積保持された注文情報に対応する暗号化ソフトウェアを復号して前記ソフト固有情報を獲得し、この獲得したソフト固有情報を前記実行器秘密情報で暗号化することによって、注文されたソフトウェアの実行権を生成し、前記ソフト実行手段は、前記注文作成手段に蓄積保持された注文情報に対応する暗号化ソフトウェアを復号してソフトウェアとソフト固有情報とを獲得し、前記実行権を実行器秘密情報を用いて復号してソフト固有情報を獲得し、これら復号によって獲得したソフト固有情報が一致している場合にのみ、復号されたソフトウェアを実行することを特徴とする、請求項1～15のいずれかに記載のソフトウェア保護システム。

【請求項17】 前記注文作成手段は、作成した注文情報を不揮発的に蓄積保持し、前記実行権生成手段が対応するソフトウェアの実行権を生成した後に当該注文情報を消去することを特徴とする、請求項1～15のいずれかに記載のソフトウェア保護システム。

【請求項18】 前記ソフト実行器および前記注文管理者は、それぞれ相互間でやりとりされた情報の履歴を保持していることを特徴とする、請求項1～15のいずれ

かに記載のソフトウェア保護システム。

【請求項19】 前記ソフト実行器および前記注文管理者は、すべてのソフトウェアの組み合わせに対応するコードをテーブルとして保持しており、前記ソフト実行器は、前記テーブルから得たコードを、前記注文情報として前記注文管理者に送付することを特徴とする、請求項1～15のいずれかに記載のソフトウェア保護システム。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、ソフトウェア保護システムに関し、より特定的には、ソフトウェアを不活性な状態で配付しておき、電話などの通信路を用いてソフトウェアを活性化するための実行権を獲得するソフトウェア保護システムに関する。

【0002】

【従来の技術】近年、種々のマルチメディア機器が開発され、ゲームや教育用のソフトウェアを始めとする多くの有償マルチメディアソフトウェアが販売されている。ところが、そのソフトウェアの保護は不完全であり、不正にコピーされたソフトウェアが数多く出回っているのが現状である。このような不正コピーを防ぐために特許法や著作権法等法律の規制があるが、同時にメカニズム面からのソフトウェア保護が強く要望されている。

【0003】例えば、フロッピーディスク等の、ソフトウェアを格納する記録媒体のフォーマットを特殊なものにすることによって、OS（オペレーティング・システム）で提供されているコピー機能では複製ができないようにする方法がある。しかしながら、このような方法でもビットごとにコピーを行なうタイプのコピーツールを用いれば、多くの場合複製が可能である。また、正規のユーザにとっては、バックアップが作れないといった不都合も生じる。

【0004】また、ソフトウェアを暗号化してコピー防止を行なう方法が提案されている。この方法では、ソフトウェアは暗号化されて不活性な状態でユーザに配付される。そして、ユーザの注文により、そのソフトウェアを特定の機器で活性化するための実行権（以下の従来例では、暗号化ファイルキー）が配付される。この実行権は、特定の機器に依存した情報なので、これを用いて他の機器で同じソフトウェアを活性化することはできない。この方法は、例えば特公平2-60007号公報に開示されている。その構成を図6に示す。なおこの図6は、特公平2-60007号公報の第1～第6図から説明上必要な部分を取り出して、1つの図にまとめ直したものである。

【0005】図6において、このソフトウェア保護システムは、ソフトウェアを実行するソフト実行器601と、ソフト実行器601に対してソフトウェアを配付する注文管理者602とを備えている。

【0006】ソフト実行器601は、暗号化ソフトウェア格納部603と、注文作成部604と、実行器ID格納部605と、実行器IDに対応した固有の実行器暗号キーを格納する暗号キー格納部606と、注文管理者602から受け取った暗号化ファイルキーを格納する暗号化ファイルキー格納部607と、暗号化ファイルキーを実行器暗号キーで復号してファイルキーを獲得する暗号化ファイルキー復号部608と、ファイルキーを用いて暗号化ソフトウェアを復号し実行するソフト復号実行部609とを含む。このような構成のソフト実行器601は、注文作成部604での注文と、実行器IDとを注文管理者602に伝えて、暗号化ソフトウェアを実行するための暗号化ファイルキーを要求する。

【0007】注文管理者602は、すべてのソフトウェアのファイルキーを格納しているファイルキー格納部610と、すべてのソフト実行器の実行器暗号キーを格納している実行器暗号キー格納部611と、ファイルキーを指定の実行器暗号キーで暗号化して暗号化ファイルキーを生成する暗号化ファイルキー生成部612とを含む。このような構成の注文管理者602は、ソフト実行器601から指定されたソフトウェアのファイルキーをファイルキー格納部610から取り出し、また実行器暗号キー格納部611から実行器IDに対応した暗号キーを取り出す。

【0008】次に、図6に示す従来のソフトウェア保護システムの動作を説明する。暗号化ソフトウェアは、ソフトウェア固有のファイルキーで元のソフトウェアを暗号化したものである。この暗号化ソフトウェアは、複数個まとめられて、予め例えばCD-ROMなどの記録媒体に格納され、ソフト実行器601に配付されている。配布された暗号化ソフトウェアは、暗号化ソフトウェア格納部603に格納されている。暗号化ソフトウェアは、これだけでは実行ができない不活性のソフトウェアである。実行を希望する場合には、注文作成部604で、注文のソフトウェアのIDを指定し、また、実行器ID格納部605に格納されているIDを注文管理者602に通知して注文を行なう。なお、この注文は、実用的には電話などを用いて行なわれる。

【0009】注文管理者602において、ファイルキー格納部610は、ソフトIDを索引としてすべてのソフトウェアのファイルキーを格納している。また、実行器暗号キー格納部611は、実行器IDを索引としてすべての実行器暗号キーを格納して管理している。そして、ソフト実行器601から注文を受け取った注文管理者602は、ファイルキー格納部610から注文のソフトウェアのファイルキーを獲得し、暗号化ファイルキー生成部612において、このファイルキーを、該当の実行器の暗号キーで暗号化して暗号化ファイルキーを作成する。そして、この暗号化ファイルキーをソフト実行器601に通知する。

【0010】暗号化ファイルキーを受け取ったソフト実行器100は、これを暗号化ファイルキー格納部107に格納しておく。そして、ソフトウェア実行時に、暗号化ファイルキー復号部108は、暗号化ファイルキー格納部107に格納された暗号化ファイルキーを、実行器暗号キー格納部106に格納された自分の暗号キーで復号して、ファイルキーを獲得する。ソフト復号実行部109は、暗号化ファイルキー復号部108で求められたファイルキーで暗号化ソフトウェアを復号し、ソフトウェアを実行する。なお、暗号化ファイルキーは、その実行器にのみ有効な情報なので、通信路や格納部から他人に獲得されても使用はできない。

【0011】

【発明が解決しようとする課題】しかしながら、上記のような従来のソフトウェア保護システムでは、ソフトウェアの注文と、暗号化ファイルキーの受け取りとがすべて電話を介して行われるものとする、以下に説明するように電話で伝える情報が多くなり、実用性に欠けたものになる。

【0012】まず、ファイルキーと実行器暗号キーのビット数について述べる。ファイルキーは、元のソフトウェアを暗号化するための鍵である。また、実行器暗号キーは、このファイルキーを暗号化するための鍵である、ところで、ここでの暗号化には、秘密鍵ブロック暗号を用いるのが一般的である。秘密鍵ブロック暗号としては、米国で最も普及している暗号方式であるDES (Data Encryption Standard)、その後日本で開発されたFEAL (Fast data Encipherment Algorithm) などが挙げられる。これらいずれの暗号化方式を採用する場合でも、安全性を確保するため、鍵のビット数は64ビット程度必要であるというのが現状での認識である。従って、ファイルキーと実行器暗号キーとは、それぞれ64ビット程度のデータとなる。なお、DESについてはFIP PUB 46, NBS Jan., 1977に、FEALについてはA. Shimizu & S. Miyaguchi: "Fast Data Encipherment Algorithm FEAL", Advances in Cryptology-EUROCRYPT'87, Springer書店に、それぞれ詳細に述べられている。

【0013】次に、暗号化ファイルキーのビット数について述べる。暗号化ファイルキーは、ファイルキーを実行器暗号キーで暗号化したものである。上述したように、ファイルキーおよび実行器暗号キーは、64ビット程度は必要であるため、この暗号化ファイルキーも64ビット程度の値となる。64ビットは、例えば10進の数字で表すとすると、ほぼ20桁くらいである。ところで、この暗号化ファイルキーは、注文管理者602を操作するオペレータから、ソフト実行器601を操作する

ユーザに対して、電話により口頭で伝えられる。さらに、ソフト実行器601のオペレータは、伝えられた暗号化ファイルキーを、ソフト実行器601に入力する必要がある。しかしながら、このように長い桁になると、言い間違え、聞き間違え、入力間違えの可能性が非常に大きくなり、実用的に問題が生じる。

【0014】逆に、上記実用性を考慮して暗号化ファイルキーの桁数を小さくすると、これに伴いファイルキーや実行暗号キーのビット数が少なくなり、安全性が低下してしまう。

【0015】また、上記従来例では、1つのソフトウェアの注文に対して1つの暗号化ファイルキーが対応する。従って、N個のソフトウェアの注文に対しては、伝える情報量はN倍となる。

【0016】なお、上記従来例では、ソフトウェアに対する暗号化ファイルキーを一旦獲得した後は、そのソフトウェアを何度でも実行できる、いわば「実行権買い取り式」になっている。しかしながら、ソフトウェアの種類によっては、ソフトウェアの使用量に応じて料金を支払う形態の方がむしろ都合が良い場合がある。この形態では、たとえば、注文管理者からうけとった実行権を用いて、所定の回数だけの実行が可能となる。

【0017】本発明の第1の目的は、安全性を劣化させずに（例えば、暗号の鍵のビット数を削減せずに）、ソフト実行器と注文管理者との間の情報量を削減することのできるソフトウェア保護システムを提供することである。

【0018】本発明の第2の目的は、ソフト実行器と注文管理者との間の情報量が、注文するソフトの個数に依存しない（すなわち、N個のソフトを注文する場合の情報量が、1個ソフトを注文する場合の情報量と同じになる）ようなソフトウェア保護システムを提供することである。

【0019】本発明の第3の目的は、回数制限等の条件付きの実行権に柔軟に対応するソフトウェア保護システムを提供することである。

【0020】

【課題を解決するための手段】請求項1に係る発明は、提供されたソフトウェアを実行する1つ以上のソフト実行器と、各ソフト実行器と通信経路を介して接続され、かつ各ソフト実行器から受けたソフトウェアの注文を管理する注文管理者とを備えたソフトウェア保護システムであって、各ソフト実行器は、固有の実行器IDを格納する実行器ID格納手段と、固有の実行器秘密情報を格納する第1の秘密情報格納手段と、1つ以上のソフトウェアの実行権の注文情報を作成して蓄積保持する注文作成手段とを含み、実行器IDおよび注文情報は、通信路を介して注文管理者に送付され、注文管理者は、各ソフト実行器に格納された全ての実行器秘密情報を格納する第2の秘密情報格納手段と、ソフト実行器から受け取っ

た実行器IDに対応する実行器秘密情報を第2の秘密情報格納手段から獲得し、この獲得した実行器秘密情報およびソフト実行器から受け取った注文情報に依存した実行権生成パスワードを生成する実行権生成パスワード生成手段とを含み、実行権生成パスワードは、通信路を介してソフト実行器に送付され、各ソフト実行器は、さらに注文作成手段に蓄積保持された注文情報と、第1の秘密情報格納手段に格納された実行器秘密情報とを用いて、注文管理者から受け取った実行権生成パスワードの正当性を検査する実行権生成パスワード検査手段と、実行権生成パスワード検査手段による検査の結果、実行権生成パスワードの正当性が確認されたときに、注文作成手段に蓄積保持された注文情報に基づいて、注文されたソフトウェアの実行権を生成する実行権生成手段と、実行権が存在するソフトウェアを、当該実行権に示される実行条件に従って実行するソフト実行手段とを含んでいる。

【0021】請求項2に係る発明は、請求項1の発明において、実行権生成パスワード生成手段は、ソフト実行器から受け取った注文情報と、第2の秘密情報格納手段から獲得した実行器秘密情報とを、出力が入力ビットのすべてに関係するようなデータ圧縮関数に入力し、このデータ圧縮関数の出力結果を、実行権生成パスワードとして出力し、実行権生成パスワード検査手段は、注文作成手段に蓄積保持された注文情報と、第1の秘密情報格納手段に格納された実行器秘密情報とを、実行権生成パスワード生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、その出力結果が注文管理者から受け取った実行権生成パスワードと一致したとき、当該実行権生成パスワードが正当であると判断することの特徴とする。

【0022】請求項3に係る発明は、請求項1の発明において、注文管理者は、公開鍵署名方式に用いられる秘密情報を格納する第3の秘密情報格納手段をさらに含み、実行権生成パスワード生成手段は、第3の秘密情報格納手段に格納された秘密情報と、第2の秘密情報格納手段から獲得した実行器秘密情報と、ソフト実行器から受け取った注文情報とを用いて、公開鍵署名方式によってデジタル署名された実行権生成パスワードを生成し、ソフト実行器は、公開鍵署名方式に用いられる秘密情報に対応した公開情報を格納する公開情報格納手段をさらに含み、実行権生成パスワード検査手段は、公開情報格納手段に格納された公開情報と、注文作成手段に蓄積保持された注文情報と、第1の秘密情報格納手段に格納された実行器秘密情報とを用いて、公開鍵署名方式に対応した署名確認方式によって、注文管理者から受け取った実行権生成パスワードの正当性を検査することの特徴とする。

【0023】請求項4に係る発明は、提供されたソフトウェアを実行する1つ以上のソフト実行器と、各ソフト

実行器と通信経路を介して接続され、かつ各ソフト実行器から受けたソフトウェアの注文を管理する注文管理者とを備えたソフトウェア保護システムであって、各ソフト実行器は、固有の実行器IDを格納する実行器ID格納手段と、固有の実行器秘密情報を格納する第1の秘密情報格納手段と、1つ以上のソフトウェアの実行権の注文情報を作成して蓄積保持する注文作成手段と、注文作成手段で注文情報が作成される毎に変化する変化値を生成して蓄積保持する変化値生成手段とを含み、実行器ID、注文情報および変化値は、通信路を介して注文管理者に送付され、注文管理者は、各ソフト実行器に格納された全ての実行器秘密情報を格納する第2の秘密情報格納手段と、ソフト実行器から受け取った実行器IDに対応する実行器秘密情報を第2の秘密情報格納手段から獲得し、この獲得した実行器秘密情報、ソフト実行器から受け取った注文情報および変化値に依存した実行権生成パスワードを生成する実行権生成パスワード生成手段とを含み、実行権生成パスワードは、通信路を介してソフト実行器に送付され、各ソフト実行器は、さらに注文作成手段に蓄積保持された注文情報と、変化値生成手段に蓄積保持された変化値と、第1の秘密情報格納手段に格納された実行器秘密情報とを用いて、注文管理者から受け取った実行権生成パスワードの正当性を検査する実行権生成パスワード検査手段と、実行権生成パスワード検査手段による検査の結果、実行権生成パスワードの正当性が確認されたときに、注文作成手段に蓄積保持された注文情報に基づいて、注文されたソフトウェアの実行権を生成する実行権生成手段と、実行権が存在するソフトウェアを、当該実行権に示される実行条件に従って実行するソフト実行手段とを含んでいる。

【0024】請求項5に係る発明は、請求項4の発明において、実行権生成パスワード生成手段は、ソフト実行器から受け取った注文情報および変化値と、第2の秘密情報格納手段から獲得した実行器秘密情報とを、出力が入力ビットのすべてに関係するようなデータ圧縮関数に入力し、このデータ圧縮関数の出力結果を、実行権生成パスワードとして出力し、実行権生成パスワード検査手段は、注文作成手段に蓄積保持された注文情報と、変化値生成手段に蓄積保持された変化値と、第1の秘密情報格納手段に格納された実行器秘密情報とを、実行権生成パスワード生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、その出力結果が注文管理者から受け取った実行権生成パスワードと一致したとき、当該実行権生成パスワードが正当であると判断することを特徴とする。

【0025】請求項6に係る発明は、提供されたソフトウェアを実行する1つ以上のソフト実行器と、各ソフト実行器と通信経路を介して接続され、かつ各ソフト実行器から受けたソフトウェアの注文を管理する注文管理者とを備えたソフトウェア保護システムであって、各ソフト

実行器は、固有の実行器IDを格納する実行器ID格納手段と、固有の実行器秘密情報を格納する第1の秘密情報格納手段と、1つ以上のソフトウェアの実行権の注文情報を作成して蓄積保持する注文作成手段と、注文作成手段で注文情報が作成される毎にタイムスタンプを生成して蓄積保持する第1のタイムスタンプ生成手段とを含み、実行器IDおよび注文情報は、通信路を介して注文管理者に送付され、注文管理者は、ソフト実行器から実行器IDおよび注文情報を受け取ると、タイムスタンプを生成する第2のタイムスタンプ生成手段と、各ソフト実行器に格納された全ての実行器秘密情報を格納する第2の秘密情報格納手段と、ソフト実行器から受け取った実行器IDに対応する実行器秘密情報を第2の秘密情報格納手段から獲得し、この獲得した実行器秘密情報と、ソフト実行器から受け取った注文情報と、第2のタイムスタンプ生成手段で生成されたタイムスタンプとに依存した実行権生成パスワードを生成する実行権生成パスワード生成手段とを含み、実行権生成パスワードは、通信路を介してソフト実行器に送付され、各ソフト実行器は、さらに注文作成手段に蓄積保持された注文情報と、第1のタイムスタンプ生成手段に蓄積保持されたタイムスタンプと、第1の秘密情報格納手段に格納された実行器秘密情報とを用いて、注文管理者から受け取った実行権生成パスワードの正当性を検査する実行権生成パスワード検査手段と、実行権生成パスワード検査手段による検査の結果、実行権生成パスワードの正当性が確認されたときに、注文作成手段に蓄積保持された注文情報に基づいて、注文されたソフトウェアの実行権を生成する実行権生成手段と、実行権が存在するソフトウェアを、当該実行権に示される実行条件に従って実行するソフト実行手段とを含んでいる。

【0026】請求項7に係る発明は、提供されたソフトウェアを実行する1つ以上のソフト実行器と、各ソフト実行器と通信経路を介して接続され、かつ各ソフト実行器から受けたソフトウェアの注文を管理する注文管理者とを備えたソフトウェア保護システムであって、各ソフト実行器は、固有の実行器IDを格納する実行器ID格納手段と、固有の実行器秘密情報を格納する第1の秘密情報格納手段と、1つ以上のソフトウェアの実行権の注文情報を作成して蓄積保持する注文作成手段と、注文情報と実行器秘密情報に依存した注文認証情報を生成して蓄積保持する注文認証情報生成手段とを含み、実行器ID、注文情報および注文認証情報は、通信路を介して注文管理者に送付され、注文管理者は、各ソフト実行器に格納された全ての実行器秘密情報を格納する第2の秘密情報格納手段と、ソフト実行器から受け取った実行器IDに対応する実行器秘密情報を第2の秘密情報格納手段から獲得し、この獲得した実行器秘密情報、ソフト実行器から受け取った注文情報および注文認証情報を用いて、ソフト実行器および注文情報の認証を行い、この認

証結果が正当である場合に、当該注文認証情報を注文を識別する注文識別情報として扱う実行器認証手段と、実行器認証手段における認証結果が正当である場合にのみ、注文情報および注文識別情報に依存した実行権生成パスワードを生成する実行権生成パスワード生成手段とを含み、実行権生成パスワードは、注文識別情報と共に、通信路を介してソフト実行器に送付され、各ソフト実行器は、さらに注文作成手段に蓄積保持された注文情報と、注文認証情報生成手段に蓄積保持された注文認証情報とを用いて、注文管理者から受け取った実行権生成パスワードの正当性を検査する実行権生成パスワード検査手段と、実行権生成パスワード検査手段による検査の結果、実行権生成パスワードの正当性が確認されたときに、注文作成手段に蓄積保持された注文情報に基づいて、注文されたソフトウェアの実行権を生成する実行権生成手段と、実行権が存在するソフトウェアを、当該実行権に示される実行条件に従って実行するソフト実行手段とを含んでいる。

【0027】請求項8に係る発明は、請求項7の発明において、注文認証情報生成手段は、注文作成手段によって作成された注文情報と、第1の秘密情報格納手段に格納された実行器秘密情報とを、出力が入力ビットのすべてに関係するようなデータ圧縮関数に入力し、このデータ圧縮関数の出力結果を、注文認証情報として出力し、実行器認証手段は、第2の秘密情報格納手段から獲得した実行器秘密情報と、ソフト実行器から受け取った注文情報とを、注文認証情報生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、このデータ圧縮関数の出力結果がソフト実行器から受け取った注文認証情報と一致する場合に、注文を行ったソフト実行器および注文情報が正当であることを認証し、実行権生成パスワード生成手段は、ソフト実行器から受け取った注文情報および注文識別情報を、注文認証情報生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、このデータ圧縮関数の出力結果を、実行権生成パスワードとして出力し、実行権生成パスワード検査手段は、注文作成手段に蓄積保持された注文情報と、注文認証情報生成手段に蓄積保持された注文認証情報とを、注文認証情報生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、その出力結果が注文管理者から受け取った実行権生成パスワードと一致したとき、当該実行権生成パスワードが正当であると判断することを特徴とする。

【0028】請求項9に係る発明は、提供されたソフトウェアを実行する1つ以上のソフト実行器と、各ソフト実行器と通信経路を介して接続され、かつ各ソフト実行器から受けたソフトウェアの注文を管理する注文管理者とを備えたソフトウェア保護システムであって、各ソフト実行器は、固有の実行器IDを格納する実行器ID格納手段と、固有の実行器秘密情報を格納する第1の秘密

情報格納手段と、1つ以上のソフトウェアの実行権の注文情報を作成して蓄積保持する注文作成手段と、注文作成手段で注文情報が作成される毎に変化し、所定のデータ構造または意味を有する変化値を生成して蓄積保持する変化値生成手段と、注文情報および実行器秘密情報に依存した値を求め、この求めた値によって変化値を変換することにより、注文認証情報を生成して蓄積保持する注文認証情報生成手段とを含み、実行器ID、注文情報および注文認証情報は、通信路を介して注文管理者に送付され、注文管理者は、各ソフト実行器に格納された全ての実行器秘密情報を格納する第2の秘密情報格納手段と、ソフト実行器から受け取った実行器IDに対応する実行器秘密情報を第2の秘密情報格納手段から獲得し、この獲得した実行器秘密情報とソフト実行器からの注文情報とに依存した値を求め、この値を用いてソフト実行器からの注文認証情報を逆変換する逆変換手段と、逆変換手段の出力が変化値と同じデータ構造または意味を有する場合に、ソフト実行器および注文情報が正当であることを認証する実行器認証手段と、実行器認証手段における認証結果が正当である場合にのみ、注文情報に依存した実行権生成パスワードを生成する実行権生成パスワード生成手段とを含み、実行権生成パスワードは、通信路を介してソフト実行器に送付され、各ソフト実行器は、さらに注文作成手段に蓄積保持された注文情報と、注文認証情報生成手段に蓄積保持された注文認証情報とを用いて、注文管理者から受け取った実行権生成パスワードの正当性を検査する実行権生成パスワード検査手段と、実行権生成パスワード検査手段による検査の結果、実行権生成パスワードの正当性が確認されたときに、注文作成手段に蓄積保持された注文情報に基づいて、注文されたソフトウェアの実行権を生成する実行権生成手段と、実行権が存在するソフトウェアを、当該実行権に示される実行条件に従って実行するソフト実行手段とを含んでいる。

【0029】請求項10に係る発明は、請求項9の発明において、注文認証情報生成手段は、注文作成手段によって作成された注文情報と、第1の秘密情報格納手段に格納された実行器秘密情報とを、出力が各入力ビットのすべてに関係するようなデータ圧縮関数に入力し、その出力結果と変化値との排他的論理和を注文認証情報として出力し、逆変換手段は、ソフト実行器からの注文情報と、第2の秘密情報格納手段から獲得した実行器秘密情報とを、注文認証情報生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、その出力結果とソフト実行器からの注文認証情報との排他的論理和を演算することにより、当該注文認証情報を逆変換し、実行権生成パスワード生成手段は、ソフト実行器から受け取った注文情報および注文識別情報を、注文認証情報生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、このデータ圧縮関数の出力結果を、実行権生成

パスワードとして出力し、実行権生成パスワード検査手段は、注文作成手段に蓄積保持された注文情報と、注文認証情報生成手段に蓄積保持された注文認証情報とを、注文認証情報生成手段で用いているデータ圧縮関数と同じデータ圧縮関数に入力し、その出力結果が注文管理者から受け取った実行権生成パスワードと一致したとき、当該実行権生成パスワードが正当であると判断することを特徴とする。

【0030】請求項11に係る発明は、請求項7または9の発明において、実行権生成パスワード生成手段は、ソフト実行器から受け取った注文情報に加えて、第2の秘密情報格納手段から獲得した実行器秘密情報に依存した実行権生成パスワードを生成し、実行権生成パスワード検査手段は、注文作成手段に蓄積保持された注文情報と、第1の秘密情報格納手段に格納されている実行器秘密情報とを用いて、注文管理者から受け取った実行権生成パスワードの正当性を検査することを特徴とする。

【0031】請求項12に係る発明は、請求項7または9の発明において、注文管理者は、公開鍵署名方式に用いられる秘密情報を格納する第3の秘密情報格納手段をさらに含み、実行権生成パスワード生成手段は、第3の秘密情報格納手段に格納された秘密情報と、第2の秘密情報格納手段から獲得した実行器秘密情報と、ソフト実行器から受け取った注文情報および注文認証情報とを用いて、公開鍵署名方式によってデジタル署名された実行権生成パスワードを生成し、ソフト実行器は、公開鍵署名方式に用いられる秘密情報に対応した公開情報を格納する公開情報格納手段をさらに含み、実行権生成パスワード検査手段は、公開情報格納手段に格納された公開情報と、注文作成手段に蓄積保持された注文情報と、注文認証情報生成手段に蓄積保持された注文認証情報と、第1の秘密情報格納手段に格納された実行器秘密情報とを用いて、公開鍵署名方式に対応した署名確認方式によって、注文管理者から受け取った実行権生成パスワードの正当性を検査することを特徴とする。

【0032】請求項13に係る発明は、提供されたソフトウェアを実行する1つ以上のソフト実行器と、各ソフト実行器と通信経路を介して接続され、かつ各ソフト実行器から受けたソフトウェアの注文を管理する注文管理者とを備えたソフトウェア保護システムであって、各ソフト実行器は、固有の実行器IDを格納する実行器ID格納手段と、固有の実行器秘密情報を格納する第1の秘密情報格納手段と、1つ以上のソフトウェアの実行権の注文情報を作成して蓄積保持する注文作成手段とを含み、実行器IDおよび注文情報は、通信路を介して注文管理者に送付され、注文管理者は、第1の実行権生成モジュールを格納する第1のモジュール格納手段と、各ソフト実行器に格納された全ての実行器秘密情報を格納する第2の秘密情報格納手段と、ソフト実行器からの注文情報と、第2の秘密情報格納手段から獲得した実行器秘

密情報とに依存した値を計算する第1の計算手段と、第1の計算手段の計算結果を用いて、第1の実行権生成モジュールを変換することにより、実行権生成パスワードを生成する実行権生成パスワード生成手段とを含み、実行権生成パスワードは、通信路を介してソフト実行器に送付され、各ソフト実行器は、さらに第2の実行権生成モジュールを格納する第2のモジュール格納手段と、注文作成手段に蓄積保持された注文情報と、第1の秘密情報格納手段に格納された実行器秘密情報とに依存した値を計算する第2の計算手段と、第2の計算手段の計算結果を用いて、注文管理者からの実行権生成パスワードを逆変換することにより、第1の実行権生成モジュールを生成する実行権生成パスワード逆変換手段と、実行権生成パスワード逆変換手段により生成された第1の実行権生成モジュールと、第2のモジュール格納手段に格納されている第2の実行権生成モジュールとを用いて、注文作成手段に蓄積保持された注文情報に基づくソフトウェアの実行権を生成する実行権生成手段と、実行権生成後、第1の実行権生成モジュールを消去するモジュール消去手段と、実行権が存在するソフトウェアを、当該実行権に示される実行条件に従って実行するソフト実行手段とを含んでいる。

【0033】請求項14に係る発明は、請求項13の発明において、第1のモジュール格納手段に格納されている第1の実行権生成モジュールは、所定のデータ構造や意味をもっており、実行権生成手段は、実行権生成パスワード逆変換手段の逆変換結果が第1の実行権生成モジュールと同じデータ構造や意味を持っている場合のみ、第1の実行権生成モジュールと、第2の実行権生成モジュールとを用いて、注文作成手段に蓄積保持された注文情報に基づくソフトウェアの実行権を生成することを特徴とする。

【0034】請求項15に係る発明は、請求項13の発明において、注文管理者は、公開鍵署名方式に用いられる秘密情報を格納する第3の秘密情報格納手段をさらに含み、実行権生成パスワード生成手段は、第1の計算手段の計算結果と、第3の秘密情報格納手段に格納された秘密情報とを用いて、第1の実行権生成モジュールを変換することにより、公開鍵署名方式によってデジタル署名された実行権生成パスワードを生成し、ソフト実行器は、公開鍵署名方式に用いられる秘密情報に対応した公開情報を格納する公開情報格納手段をさらに含み、実行権生成パスワード逆変換手段は、第2の計算手段の計算結果と、公開情報とを用い、公開鍵署名方式に対応した署名確認方式によって、注文管理者からの実行権生成パスワードを逆変換することを特徴とする。

【0035】請求項16に係る発明は、請求項1～15のいずれかに記載の発明において、ソフト実行器に提供されるソフトウェアは、ソフト固有情報を含んで暗号化されており、実行権生成手段は、実行権生成パスワード

検査手段が注文管理者からの実行権生成パスワードの正当性を確認したとき、注文作成手段に蓄積保持された注文情報に対応する暗号化ソフトウェアを復号してソフト固有情報を獲得し、この獲得したソフト固有情報を実行器秘密情報で暗号化することによって、注文されたソフトウェアの実行権を生成し、ソフト実行手段は、注文作成手段に蓄積保持された注文情報に対応する暗号化ソフトウェアを復号してソフトウェアとソフト固有情報とを獲得し、実行権を実行器秘密情報を用いて復号してソフト固有情報を獲得し、これら復号によって獲得したソフト固有情報が一致している場合にのみ、復号されたソフトウェアを実行することを特徴とする。

【0036】請求項17に係る発明は、請求項1～15のいずれかに記載の発明において、注文作成手段は、作成した注文情報を不揮発的に蓄積保持し、実行権生成手段が対応するソフトウェアの実行権を生成した後に当該注文情報を消去することを特徴とする。

【0037】請求項18に係る発明は、請求項1～15のいずれかに記載の発明において、ソフト実行器および注文管理者は、それぞれ相互間でやりとりされた情報の履歴を保持していることを特徴とする。

【0038】請求項19に係る発明は、請求項1～15のいずれかに記載の発明において、ソフト実行器および注文管理者は、すべてのソフトウェアの組み合わせに対応するコードをテーブルとして保持しており、ソフト実行器は、テーブルから得たコードを、注文情報として注文管理者に送付することを特徴とする。

【0039】

【作用】請求項1に係る発明では、ソフト実行器は、実行器IDおよび注文情報を生成し、これらを注文管理者に送付することにより、ソフトウェアの注文を行なう。注文管理者は、受け取った注文情報と、実行IDに対応する実行器秘密情報とに依存した情報である実行権生成パスワードを生成し、ソフト実行器に送付する。ソフト実行器は、この実行権生成パスワードが、注文管理者に先に送付した注文情報や、自分自身の実行器秘密情報と整合している否かを判断し、整合している場合は、注文情報に対応したソフトウェアの実行権を生成し、当該ソフトウェアを実行する。このように、注文管理者からソフト実行器には、実行権生成パスワード、すなわちソフト実行器が注文情報に対応したソフト実行権を生成するための許可を与える情報だけが送付されるため、実行権そのものを送付する場合に比べて、桁数を削減できる。また、ソフト実行器が複数のソフトウェアの注文を1回に行なっても、その情報量は1つのソフトウェアを注文した場合の情報量と同じになる。

【0040】請求項2に係る発明では、注文管理者は、ソフト実行器から受け取った注文情報と、実行器IDに対応する実行器秘密情報とを、所定のデータ圧縮関数（例えば、ハッシュ関数）に入力し、このデータ圧縮関

数の出力結果を、実行権生成パスワードとして出力する。一方、ソフト実行器は、内部に蓄積保持された注文情報と、自分自身の実行器秘密情報とを、上記と同様のデータ圧縮関数に入力し、その出力結果が注文管理者から受け取った実行権生成パスワードと一致したとき、当該実行権生成パスワードが正当であると判断する。このように、データ圧縮関数を用いて実行権生成パスワードが生成されるので、その桁数をより一層削減できる。

【0041】請求項3に係る発明では、注文管理者は、公開鍵署名方式に用いられる秘密情報を保持し、この秘密情報と、実行器IDに対応する実行器秘密情報と、ソフト実行器から受け取った注文情報とを用いて、公開鍵署名方式によってデジタル署名された実行権生成パスワードを生成する。一方、ソフト実行器は、公開鍵署名方式に用いられる秘密情報に対応した公開情報を保持し、この公開情報と、注文情報と、自分自身の実行器秘密情報とを用いて、公開鍵署名方式に対応した署名確認方式によって、注文管理者から受け取った実行権生成パスワードの正当性を検査する。このように、ソフト実行器には、デジタル署名された実行権生成パスワードが送付されるので、ソフト実行器側で実行権生成パスワードを偽造することがほとんど不可能となり、極めて安全性の高いソフトウェア保護システムが実現できる。

【0042】請求項4に係る発明では、注文情報を作成する毎に変化する変化値が注文情報と共に、ソフト実行器から注文管理者に送付される。そして、注文管理者は、実行器秘密情報、注文情報および変化値に依存した実行権生成パスワードを生成し、注文管理者に送付する。ソフト実行器は、注文情報、変化値および実行器秘密情報を用いて、実行権生成パスワードの正当性を検査する。従って、注文管理者からの実行権生成パスワードは、対応する変化値を保持するソフト実行器でのみ有効となる。そのため、異なる時期に同じ注文を作成し、以前の実行権生成パスワードを入力しても、後の注文時に生成される変化値は、以前の注文時に生成される変化値とは異なっているため、入力したパスワードが有効にはならない。

【0043】請求項5に係る発明では、注文管理者は、注文情報、変化値および実行器秘密情報を、ハッシュ関数等のデータ圧縮関数に入力し、このデータ圧縮関数の出力結果を、実行権生成パスワードとして出力する。一方、ソフト実行器は、内部に保持している注文情報、変化値および実行器秘密情報を、上記と同様のデータ圧縮関数に入力し、その出力結果が注文管理者から受け取った実行権生成パスワードと一致したとき、当該実行権生成パスワードが正当であると判断する。このように、データ圧縮関数を用いて実行権生成パスワードが生成されるので、その桁数をより一層削減できる。

【0044】請求項6に係る発明では、請求項4における変化値の代わりにタイムスタンプを用いている。この

タイムスタンプは、ソフト実行器と注文管理者とに共通に保持されているため、請求項4のようにソフト実行器から注文管理者に変化値を送付する必要がなくなる。

【0045】請求項7に係る発明では、注文情報および実行器秘密情報に依存した注文認証情報を導入することにより、注文管理者側で実行権生成パスワードの発行に先だって、注文情報とソフト実行器の認証を行うことができる。そして、注文管理者は、認証結果がOKの場合には、上記注文認証情報を注文を識別する注文識別情報として扱い、これら注文情報および注文識別情報に依存した実行権生成パスワードを生成してソフト実行器に送付する。ソフト実行器は、内部に保持している注文情報および注文認証情報を用いて、注文管理者から受け取った実行権生成パスワードの正当性を検査し、この検査結果がOKのときに、ソフトウェアの実行権を生成し、対応するソフトウェアを実行する。

【0046】請求項8に係る発明では、ソフト実行器は、注文情報および実行器秘密情報を、ハッシュ関数等のデータ圧縮関数に入力し、このデータ圧縮関数の出力結果を、注文認証情報として得ている。また、注文管理者は、実行器秘密情報と、ソフト実行器から受け取った注文情報とを、上記と同様のデータ圧縮関数に入力し、その出力結果がソフト実行器から受け取った注文認証情報と一致する場合に、ソフト実行器および注文情報が正当であることを認証する。さらに、注文管理者は、注文情報および注文識別情報を、上記と同様のデータ圧縮関数に入力し、このデータ圧縮関数の出力結果を、実行権生成パスワードとして出力し、ソフト実行器に送付する。一方、ソフト実行器は、注文情報および注文認証情報を、上記と同様のデータ圧縮関数に入力し、その出力結果が注文管理者からの実行権生成パスワードと一致したとき、当該実行権生成パスワードが正当であると判断する。

【0047】請求項9に係る発明では、ソフト実行器は、注文情報の作成毎に変化しかつ所定のデータ構造または意味を有する変化値を、注文情報および実行器秘密情報に依存した値によって変換することにより、注文認証情報を生成する。この注文認証情報は、実行器IDおよび注文情報と共に、注文管理者に送付される。注文管理者は、実行器IDに対応する実行器秘密情報と、ソフト実行器からの注文情報とに依存した値を求め、この値を用いてソフト実行器からの注文認証情報を逆変換し、この逆変換結果が上記変化値と同じデータ構造または意味を有する場合に、ソフト実行器および注文情報が正当であることを認証する。そして、注文管理者は、この認証結果が正当である場合にのみ、注文情報に依存した実行権生成パスワードを生成してソフト実行器に送付する。ソフト実行器は、内部に保持している注文情報および注文認証情報を用いて、注文管理者から受け取った実行権生成パスワードの正当性を検査し、実行権生成パス

ワードの正当性が確認されたときに、注文されたソフトウェアの実行権を生成する。このような構成により、ソフト実行器と注文管理者間でやりとりされる情報量を増やすことなく、回数制限機能およびユーザ認証機能を実現できる。

【0048】請求項10に係る発明では、ソフト実行器は、注文情報および実行器秘密情報を、ハッシュ関数等のデータ圧縮関数に入力し、その出力結果と変化値との排他的論理和を注文認証情報として得ている。注文管理者は、ソフト実行器からの注文情報と、実行器IDに対応する実行器秘密情報とを、上記と同様のデータ圧縮関数に入力し、その出力結果とソフト実行器からの注文認証情報との排他的論理和を演算することにより、当該注文認証情報を逆変換している。また、注文管理者は、ソフト実行器からの注文情報および注文識別情報を、上記と同様のデータ圧縮関数に入力し、このデータ圧縮関数の出力結果を、実行権生成パスワードとして出力するようにしている。ソフト実行器は、内部に保持している注文情報および注文認証情報を、上記と同様のデータ圧縮関数に入力し、その出力結果が注文管理者から受け取った実行権生成パスワードと一致したとき、当該実行権生成パスワードが正当であると判断する。

【0049】請求項11に係る発明では、注文管理者は、ソフト実行器から受け取った注文情報と、実行器IDに対応する実行器秘密情報とに依存した実行権生成パスワードを生成する。一方、ソフト実行器は、内部に保持している注文情報および実行器秘密情報を用いて、注文管理者から受け取った実行権生成パスワードの正当性を検査する。

【0050】請求項12に係る発明では、注文管理者は、公開鍵署名方式に用いられる秘密情報と、実行器IDに対応する実行器秘密情報と、ソフト実行器から受け取った注文情報および注文認証情報とを用いて、公開鍵署名方式によってデジタル署名された実行権生成パスワードを生成し、ソフト実行器に送付する。ソフト実行器は、公開鍵署名方式に用いられる秘密情報に対応した公開情報と、内部に保持している注文情報、注文認証情報および実行器秘密情報とを用い、公開鍵署名方式に対応した署名確認方式によって、注文管理者から受け取った実行権生成パスワードの正当性を検査する。このように、ソフト実行器には、デジタル署名された実行権生成パスワードが送付されるので、ソフト実行器側で実行権生成パスワードを偽造することがほとんど不可能となり、極めて安全性の高いソフトウェア保護システムが実現できる。

【0051】請求項13に係る発明では、ソフト実行器側に格納された第2の実行権生成モジュールだけでは、実行権の生成はできない。注文管理者は、この第2の実行権生成モジュールを活性化させるための第1の実行権生成モジュールを、ソフト実行器からの注文情報と実行

器秘密情報に依存した形に変換して実行権生成パスワードとして送付する。ソフト実行器側では、正しい注文情報と実行器である場合、この実行権生成パスワードを用いて、内部に保持している第2の実行権生成モジュールを活性化させることができる。この場合、ソフト実行器が不正に実行権を生成しようとしても、情報量的に第2の実行権生成モジュールにそのための情報が不足しているため不正はできない。そのため、より一層安全なソフトウェア保護システムが実現できる。

【0052】請求項14に係る発明では、注文管理者に格納されている第1の実行権生成モジュールは、所定のデータ構造や意味をもっており、ソフト実行器は、実行権生成パスワードの逆変換結果が第1の実行権生成モジュールと同じデータ構造や意味を持っている場合にのみ、第1の実行権生成モジュールと、第2の実行権生成モジュールとを用いて、ソフトウェアの実行権を生成する。

【0053】請求項15に係る発明では、注文管理者は、公開鍵署名方式に用いられる秘密情報と、注文情報および実行器秘密情報に依存した値である第1の計算手段の計算結果とを用いて、第1の実行権生成モジュールを変換することにより、公開鍵署名方式によってデジタル署名された実行権生成パスワードを生成する。一方、ソフト実行器は、公開鍵署名方式に用いられる秘密情報に対応した公開情報と、注文情報および実行器秘密情報に依存した値である第2の計算手段の計算結果とを用い、公開鍵署名方式に対応した署名確認方式によって、注文管理者からの実行権生成パスワードを逆変換する。

【0054】請求項16に係る発明では、ソフト実行器に提供されるソフトウェアは、ソフト固有情報を含んで暗号化されている。そして、ソフト実行器は、注文管理者からの実行権生成パスワードの正当性を確認したとき、内部に保持している注文情報に対応する暗号化ソフトウェアを復号してソフト固有情報を獲得し、この獲得したソフト固有情報を実行器秘密情報で暗号化することによって、注文されたソフトウェアの実行権を生成する。また、ソフト実行器は、内部に保持している注文情報に対応する暗号化ソフトウェアを復号してソフトウェアとソフト固有情報とを獲得し、実行権を実行器秘密情報を用いて復号してソフト固有情報を獲得し、これら復号によって獲得したソフト固有情報が一致している場合にのみ、復号されたソフトウェアを実行する。

【0055】請求項17に係る発明では、ソフト実行器は、作成した注文情報を不揮発的に蓄積保持し、対応するソフトウェアの実行権を生成した後に当該注文情報を消去する。これによって、とりあえず注文だけを連続して送付し、後に返送されてくる実行権生成パスワードに基づいて、ソフトウェアの実行を制御することができる。

【0056】請求項18に係る発明では、ソフト実行器

および注文管理者は、それぞれ相互間でやりとりされた情報の履歴を保持している。これによって、後に生じるトラブル等に柔軟に対処できる。

【0057】請求項19に係る発明では、ソフト実行器および注文管理者は、すべてのソフトウェアの組み合わせに対応するコードをテーブルとして保持しており、ソフト実行器は、テーブルから得たコードを、注文情報として注文管理者に送付するようにしている。これによって、注文情報のデータ量をより一層削減できる。

【0058】

【実施例】

(1) 第1の実施例

図1は、本発明の第1の実施例に係るソフトウェア保護システムの構成を示すブロック図である。図1において、本実施例のソフトウェア保護システムは、ソフトウェアを実行するソフト実行器101と、ソフト実行器101に対してソフトウェアを配付する注文管理者102とを備えている。

【0059】ソフト実行器101は、暗号化ソフトウェア格納部3と、注文作成部4と、実行器ID格納部5と、実行器IDに対応した実行器暗号キー（実行器秘密情報の一例）を格納する実行器暗号キー格納部6と、注文管理者102から受け取った実行権生成パスワードの正当性を検査する実行権生成パスワード検査部7と、注文情報に基づいて実行器暗号キーを使用して実行権を生成する実行権生成部8と、生成した実行権を格納する実行権格納部9と、ソフトウェア実行時に対応する実行権が有効である場合に上記暗号化ソフトウェアを復号して実行するソフト復号実行部10とを含む。このような構成のソフト実行器101において、注文作成部4で作成された注文情報と、実行器IDとは、注文管理者102に伝えられ、暗号化ソフトウェアを実行するための実行権生成パスワードが要求される。

【0060】一方、注文管理者102は、すべてのソフト実行器の実行器暗号キーを格納している実行器暗号キー格納部11と、注文情報および該当のソフト実行器の実行器暗号鍵に依存した実行権生成パスワードを生成する実行権生成パスワード生成部12とを含む。

【0061】次に、図1に示す第1の実施例のソフトウェア保護システムの動作を説明する。なお、以下の説明では、ソフト実行器101と注文管理者102との間の通信は、それらを操作している人間が電話を介して行なうものとする。

【0062】まず、暗号化ソフトウェアの生成方法について説明する。ソフトウェアSoftAに対して固有の認証子authAが定められる。そして、SoftAと、認証子authAとが結合され、この結合されたデータがシステム内で共通の秘密情報S（図示せず）を用いて暗号化され、暗号化ソフトウェアが生成される。式で表現すると、暗号化ソフトESoftAは、

$E\text{Soft}A = E(S, \text{auth}A \parallel \text{Soft}A)$ となる。なお、上式において、 $E(S, *)$ は S を鍵とした暗号関数を、 \parallel は結合を示す。同様に、他のソフトウェア $\text{Soft}B$ に関しても、固有の認証子 $\text{auth}B$ を対応させて暗号化ソフトウェアが作成される。以下、同じような手続きで作成された複数の暗号化ソフトウェアが、例えば1枚のCD-ROMに格納され、予めソフト実行器101に配付される。図1の実施例においては、この暗号化ソフトウェアは、暗号化ソフトウェア格納部3に格納されている。

【0063】ソフト実行器101のユーザは、ソフト実行器101を操作することにより、暗号化ソフトウェア格納部3に格納されているソフトウェアの中から、実行することを希望するソフトウェアを指定する。例えば、ソフトウェアAとソフトウェアCという具合にである。応じて、注文作成部4は、対応する注文情報を生成する。この注文情報は、注文作成部4内に一旦蓄積される。そして、ソフト実行器101のユーザは、この注文情報と、ソフト実行器101に固有の実行器ID（実行器ID格納部5に格納されている）とを、注文管理者102のオペレータに電話で通知する。これによって、注文が完了する。

【0064】注文管理者102内の実行権生成パスワード生成部12は、ソフト実行器101から通知された注文情報が入力されると、まず注文を出したソフト実行器101の認証（または、ユーザ認証）を行ない、次に、実行器暗号キー格納部11から該当のソフト実行器101の暗号キーを獲得する。次に、実行権生成パスワード生成部12は、ソフト実行器101から受け取った注文情報と、実行器暗号キー格納部11から獲得した実行器暗号キーとを結合して、この結合データを予め定められたハッシュ関数に入力する。このハッシュ関数の出力は、実行権生成パスワードとして、電話でソフト実行器101のユーザに伝えられる。

【0065】ここで、ハッシュ関数は、データ圧縮型暗号処理であり、その出力が入力のすべてのビットに依存しており、出力が同じとなる異なる入力ペアを容易には見付けることができないという性質を持っている。その具体的な構成方法については、例えば池野、小山共著の「現代暗号理論」、電子通信学会発行の224ページから225ページに詳しく記述されている。なお、本実施例で用いるハッシュ関数は、例えば入力を10進10桁くらいの値に圧縮する関数とする。

【0066】ソフト実行器101のユーザは、電話で受け取った実行権生成パスワードを、実行権生成パスワード検査部7に入力する。この実行権生成パスワード検査部7には、注文管理者102で用いているのと同じハッシュ関数が格納されている。そして、実行権生成パスワード検査部7は、このハッシュ関数に、注文作成部4内に蓄積されている注文情報と、実行器暗号キー格納部6

内に格納されている実行器暗号キーとを結合したデータを入力する。次に、実行権生成パスワード検査部7は、このハッシュ関数の出力結果を、注文管理者102から得た実行権生成パスワードと比較する。このとき、注文作成部4内に蓄積されている注文情報が実際に注文管理者102に伝えられた内容と違っていたり、ソフト実行器101内に格納されている実行器暗号キーが注文管理者102内で獲得された実行器暗号キーと異なっている場合には、この比較結果は、NG（不一致）となる。

【0067】実行権生成部8は、上記実行権生成パスワード検査部7での検査結果がOK（一致）のときにのみ、起動されて実行権の生成を行なう。その方法は、次のとおりである。まず、実行権生成部8は、注文作成部4内に蓄積されている注文情報を参照して、対応する暗号化ソフトウェアを暗号化ソフト格納部3から取り出す。例えば、暗号化ソフトウェアとして、 $E\text{Soft}A$ を取り出したと想定すると、実行権生成部8は、この取り出した暗号化ソフトウェア $E\text{Soft}A$ を、システム内で共通の秘密鍵 S （図示せず）を用いて復号し、対応する認証子 $\text{auth}A$ を得る。次に、実行権生成部8は、この認証子 $\text{auth}A$ を、実行器暗号キー格納部6に格納されている実行器暗号キーを用いて暗号化し、この暗号化で得た認証子を、実行権として実行権格納部9に格納する。一般的に、特定の実行器暗号キー Sx を備えたソフト実行器 X の場合、ソフト A の実行権は、 $E(Sx, \text{auth}A)$ となる。この実行権は、実行器 X 固有の暗号キー Sx で暗号化されているため、実行権格納部9として、外部からの読出プロテクトが施されていない一般のメモリを用いた場合でも、機密漏洩の問題は生じない。

【0068】ソフトウェア A を実行する際には、ソフト復号実行部10は、まず暗号化ソフトウェア $E\text{Soft}A$ をシステム内で共通の秘密鍵 S を用いて復号し、認証子 $\text{auth}A$ と $\text{Soft}A$ とを得る。次に、ソフト復号実行部10は、ソフトウェア A に対応する実行権 $E(Sx, \text{auth}A)$ を、実行権格納部9から取り出す。次に、ソフト復号実行部10は、実行権 $E(Sx, \text{auth}A)$ を実行器暗号キー Sx で復号して得た値と、先に得ていた認証子とを比較する。そして、ソフト復号実行部10は、両者が一致するときに、ソフトウェア $\text{Soft}A$ を実行する。

【0069】以上説明したように、本発明の第1の実施例では、従来システムにおいて注文管理者が行なっていた実行権の生成（従来例では、暗号化ファイルキーの生成に相当する）を、ソフト実行器側で行なっている。すなわち、本実施例では、注文管理者102は、実行権生成部8に対して起動の許可を与えるだけの機能を有している。ここで、許可情報となる実行権生成パスワードは、注文情報と実行器の暗号キーとに依存したハッシュ関数の出力であるため、10進10桁程度の値で実現で

き、電話での伝え間違い、入力間違いが少なくなり、ユーザインタフェースも良好になる。また、注文情報および実行器の認証が可能となる。また、注文管理者102は、実行権生成部8に対し起動の許可を与えるだけなので、実行権生成パスワードは、注文するソフトの個数に依存せず、いつも少ない桁数で実現できる。

【0070】次に、上記第1の実施例のソフトウェア保護システムにおいて、不正行為に対する安全性について説明する。第1の実施例のシステムでは、注文管理者102がソフト実行器101から注文を受けて、それに対する実行権生成パスワードを発行する際に、注文に対応した課金が行なわれるものとしている。支払の方法としては、一般的にはクレジット等の方法が採用されるであろう。従って、ソフト実行器101において、課金の対象とならない不正な実行権を勝手に生成することが困難であれば、システムの安全性が高いということになる。

【0071】まず、不正な実行権を生成させるための外部からの攻撃として、ソフト実行器101内の実行権生成部8を単独で動作させることが考えられる。また、注文作成部4で作成した注文以外のソフトウェアの実行権を、実行権生成部8で作成させるという攻撃も考えられる。これらの攻撃に対しては、ユーザがソフト実行器101内の各構成要素を変更できないようにすることにより対処できる。

【0072】第1の実施例では、実行権生成部8は、注文作成部4において注文が行なわれ、次にそれに対応する実行権生成パスワードの検査が通るという一連の作業を経た後でなければ起動しない。すなわち、第1の実施例では、実行権生成部8だけが単独で動作するように変更することはできない。また、第1の実施例では、上記一連の作業から注文作成を除いて、実行権生成パスワードの検査と実行権生成部8だけを動作させることもできない。さらに、注文作成部4において作成された注文情報が、実行権生成パスワード検査部7と実行権生成部8とで用いられる際、そのいずれにおいてもユーザは、注文情報の内容を変更できない。従って、第1の実施例では、上記のような不正な攻撃が行われても、実行権が生成されることはない。

【0073】なお、ソフト実行器101が専用器である場合や、ソフト実行器内のプログラムがCD-ROMなどで実現されている場合が非常に一般的であるということとを考慮すると、ユーザがソフト実行器内の各構成要素を変更できないようにすることは、現実には即したものである。

【0074】また、ソフトウェアを不正に実行するために、ある注文に対して受け取った実行権生成パスワードを、別の注文に対する実行権生成パスワードとして入力したり、また注文を出したソフト実行器以外のソフト実行器に入力してみるという攻撃も考えられる。ところが、第1の実施例のシステムでは、実行権生成パスワー

ドは、注文情報と実行器暗号キーとに依存した値であるため、注文情報やソフト実行器が違う場合には、実行権生成パスワードの検査が通らない。従って、第1の実施例は、このような不正な攻撃に対しても対処できる。

【0075】さらに、ソフトウェアを不正に実行するために、ある注文の実行権生成パスワードに対し、総当たり的に適当な値のパスワードを入力してみるという攻撃も考えられる。しかしながら、第1の実施例では、パスワードの桁数を10進で10桁くらいに設定しているため、不正ユーザが適当にパスワードを入力してみても、それがたまたま検査部7をパスする確率は、實際上0に近いと考えられる。従って、第1の実施例は、このような攻撃に対しても対処できる。

【0076】さらに、ソフトウェアを不正に実行するために、ユーザが実行権生成パスワードを偽造することも考えられる。これに対しては、例えば公開鍵署名法を用いて対処することができる。公開鍵署名法を用いた場合、実行権生成パスワードを生成するためには、注文管理者102の秘密鍵が必要となるため、一般のユーザには、パスワードの偽造はできない。また、実行権生成パスワードを発行するための手続き（演算）を公開しないという方法でも対処できる。上記第1の実施例では、ソフト実行器2内の実行権生成パスワード検査部7は、実行権生成パスワードの生成と同じ手順で検査を行なうが、ユーザには実行権生成パスワード検査部7のアルゴリズムが公開されない。さらに、実行権生成パスワード検査部7が単独で使用する（すなわち、ソフト実行器101内の構成要素を変更する）ことができないため、パスワード偽造の問題は生じない。

【0077】また、実行権格納部9を、読出プロテクト機能を持たない一般のメモリを用いて構成した場合、実行権格納部9に格納されている実行権を解析することにより、実行権を不正に生成することも考えられる。しかしながら、第1の実施例では、実行権格納部9内に格納されている実行権は、対応するソフト実行器101の暗号キーで暗号化され、しかも実行器暗号キーが64ビット程度に設定されているので、容易には実行権を解析できない。さらに、ユーザが実行器暗号キーを観察できないようにする、またはユーザに対して復号処理を公開しないようにすることで、より安全性を高めることができる。

【0078】以上の説明により、ソフト実行器101側で実行権を偽造することは、事実上不可能であることが分かる。

【0079】なお、上記第1の実施例のシステムは、ユーザが実行権を手に入れたら、その実行権を無限回数使用できるようなシステム、すなわち「実行権買い取り」のシステムとして構成されている。従って、同じ注文情報に対する実行権生成パスワードが複数回入力されてもかまわない。

【0080】（2）第2の実施例

図2は、本発明の第2の実施例に係るソフトウェア保護システムの構成を示すブロック図である。この第2の実施例は、第1の実施例のような「実行権買い取り式」のシステムとは異なり、ソフトウェアの実行権に例えば実行回数などの条件が付加されている場合に有効となる。

【0081】ところで、注文したソフトウェアの実行回数を制限するためには、実行権生成パスワードは、最初の入力時にのみ有効とならなければならない。これは2回目以降の入力時にも有効であれば、次のような不都合が生じるからである。例えば、あるソフト実行器がソフトウェアAをN回実行する注文を作成し、注文管理者から当該注文に対する実行権生成パスワードを手に入れたとする。ソフト実行器にこのパスワードが入力されると、ソフトウェアAをN回実行できる実行権が生成される。次に、この実行権が少なくなった時点で、同じソフト実行器において再度ソフトウェアAをN回実行する注文を作成し、注文管理者には注文せずに、以前のパスワードを入力する。このパスワードがもし有効となり、再度ソフトウェアAのN回の実行権が生成されると、実質的に実行回数の制限は意味がなくなってしまう。

【0082】そこで、第2の実施例では、実行権生成パスワードを最初の入力時だけ有効とするために、第1の実施例の構成に加えて、注文ごとに変化する値を導入している。すなわち、第2の実施例では、個々の注文情報に対して、その都度値が変化し変化値を生成し、この変化値に対応した実行権生成パスワードだけが有効となるようにしている。この注文ごとに変化する変化値の実現方法としては種々考えられるが、第2の実施例では、注文ごとに乱数を発生し、この乱数を注文ごとに変化する変化値として用いるようにしている。

【0083】図2において、本実施例のソフトウェア保護システムは、ソフトウェアを実行するソフト実行器201と、ソフト実行器201に対してソフトウェアを配付する注文管理者202とを備えている。ソフト実行器201は、図1の実施例と同様の構成の、暗号化ソフトウェア格納部3と、注文作成部4と、実行器ID格納部5と、実行器暗号キー格納部6と、実行権生成部8と、実行権格納部9と、ソフト復号実行部10とを含む。さらに、ソフト実行器201は、注文のたびに起動する乱数生成器20と、受け取った実行権生成パスワードの正当性を、注文情報、実行器暗号キーおよび乱数データを用いて確かめる実行権生成パスワード検査部21とを含む。

【0084】一方、注文管理者202は、図1の実施例と同様の構成の実行器暗号キー格納部11を含む。さらに、注文管理者202は、ソフト実行器201から送付された注文情報および乱数と、実行器暗号キー格納部11から獲得したソフト実行器202の暗号キーとを用いて、それらすべてに依存する実行権生成パスワードを生

成する実行権パスワード生成部22を含む。

【0085】以上述べたように、この第2の実施例では、注文のたびに新しい乱数が生成される。そして、ソフト実行器201のユーザは、第1の実施例における注文情報および実行器IDに加え、この乱数を、注文管理者202のオペレータに電話で通知する。なお、この乱数は、ユーザインタフェースを良好にするために、10進10桁程度の値とする。

【0086】注文管理者202における実行権生成パスワード生成部22は、第1の実施例で用いたのと同様のハッシュ関数に、注文情報、乱数および該当するソフト実行器201の実行器暗号キーを結合したデータを入力する。そして、このハッシュ関数の出力結果である10進10桁の値は、実行権生成パスワードとして、ソフト実行器201のユーザに電話で通知される。

【0087】通知を受けたソフト実行器201のユーザは、上記実行権生成パスワードを、ソフト実行器201の実行権生成パスワード検査部21に入力する。実行権生成パスワード検査部21は、注文作成部4および乱数発生部20から得た注文情報および乱数と、注文管理者202から得た実行器暗号キーとを結合して、注文管理者202で使用しているハッシュ関数と同様のハッシュ関数に入力する。そして、実行権生成パスワード検査部21は、このハッシュ関数の出力結果を、入力された実行権生成パスワードと比較し、両者が一致しているときのみ、実行権生成部8を起動する。これ以降の動作は、前述した第1の実施例と同様であるので、その説明を省略する。

【0088】上記第2の実施例では、注文のたびに新しい乱数が自動的に生成される。そして、正規の手続きの場合、注文管理者202からは、この乱数（＝注文ごとに変化する値）に依存した実行権生成パスワードが発行される。そして、この実行権作成パスワードは、対応する乱数を保持するソフト実行器201においてのみ有効となる。従って、2回目以降の注文発生時における乱数が、1回目の注文発生時に生成された乱数とは異なるため、1回目の注文発生時に得た実行権生成パスワードを、複数回不正使用することはできない。なお、第2の実施例では、第1の実施例において説明した安全性確保に対する対処に加え、上記乱数をユーザが勝手に変更したり設定できないようにすることが好ましい。

【0089】なお、上記第2の実施例では、新しい注文が作成されるまでは、以前の乱数が残っているように構成されているが、一旦注文をセーブすることを可能とする場合には、実行権を生成した段階でそれに関与する乱数をクリアまたは更新するようにしてもよい。これによって、より一層安全性が高まる。

【0090】（3）第3の実施例

図3は、本発明の第3の実施例に係るソフトウェア保護システムの構成を示すブロック図である。この第3の実

施例は、前述の第1の実施例に効率のよい実行器認証を付加したものである。図3において、第3の実施例のソフトウェア保護システムは、ソフトウェアを実行するソフト実行器301と、ソフト実行器301に対してソフトウェアを配付する注文管理者302とを備えている。

【0091】ソフト実行器301は、前述の第1の実施例と同様の構成の、暗号化ソフトウェア格納部3と、注文作成部4と、実行器ID格納部5と、実行器暗号キー格納部6と、実行権生成部8と、実行権格納部9と、ソフト復号実行部10とを含む。さらに、ソフト実行器301は、注文情報と実行器暗号キーに依存した注文認証情報を生成する注文認証情報生成部30と、注文管理者302から受け取った実行権生成パスワードの正当性を、注文情報、実行器暗号キーおよび注文認証情報を用いて検査する実行権生成パスワード検査部31とを含む。

【0092】一方、注文管理者302は、図1の実施例と同様の構成の実行器暗号キー格納部11を含む。さらに、注文管理者302は、ソフト実行器301から受け取った注文情報および注文認証情報と、実行器暗号キー格納部11から獲得した実行器暗号キーとを用いてソフト実行器の認証を行なう実行器認証部32と、実行器認証結果がOKの場合に、注文認証情報を注文情報を識別する注文識別情報として用い、さらに注文情報と注文識別情報（＝注文認証情報）とを用いて実行権生成パスワードを生成する実行権生成パスワード生成部33とを含む。

【0093】次に、図3に示す第3の実施例のソフトウェア保護システムの動作を説明する。まず、ソフト実行器301の注文作成部4で注文情報が作成されると、注文認証情報生成部30は、第1の実施例で用いたのと同様のハッシュ関数に、注文情報と実行器暗号キー格納部6から得た実行器暗号キーとを結合したデータを入力する。そして、ソフト実行器301のユーザは、上記ハッシュ関数の出力結果を注文認証情報として、注文情報および実行器IDに加えて、注文管理者302のオペレータに電話で伝える。なお、注文情報および注文認証情報は、それぞれ、注文作成部4および注文認証情報生成部30に蓄積されている。

【0094】注文管理者302では、まず実行器認証部32が、ソフト実行器301から受け取った実行器IDを鍵として、実行器暗号キー格納部11を検索し、該当するソフト実行器301の実行器暗号キーを獲得する。次に、実行器認証部32は、ソフト実行器301から受け取った注文情報と実行器暗号キーとを結合したデータを、ソフト実行器301で用いているのと同様のハッシュ関数に入力する。次に、実行器認証部32は、このハッシュ関数の出力結果を、ソフト実行器301から受け取った注文認証情報と比較する。そして、実行器認証部32は、比較結果が一致しているときのみ、相手が正し

い実行器であり正しい注文を行なったものと判断し、注文認証情報を注文を識別する注文識別情報として、例えば注文情報およびこれに関連した情報の管理や授受に用いる。また、実行権生成パスワード生成部33は、注文情報と注文識別情報（＝注文認証情報）とを結合したデータを、上記と同様のハッシュ関数に入力する。そして、注文管理者302のオペレータは、そのハッシュ関数の出力結果を、実行権生成パスワードとして、注文識別情報と対してソフト実行器301のユーザに電話で伝える。

【0095】ソフト実行器301のユーザは、伝えられた実行権生成パスワードを、実行権生成パスワード検査部31に入力する。応じて、実行権生成パスワード検査部31は、注文作成部4に蓄積された注文情報と、注文認証情報生成部30に蓄積された注文認証情報とを結合したデータをハッシュ関数に入力し、その出力結果が入力された実行権生成パスワードと一致するか否かを検査する。そして、実行権生成パスワード検査部31は、この検査結果が一致している場合にのみ、実行権生成部8を起動する。これ以降の動作は、第1の実施例と同様であり、その説明を省略する。

【0096】上記第3の実施例では、注文管理者302は、実行権生成パスワードの発行前に、注文情報およびソフト実行器の認証を行なう。実際のシステムでは、注文により課金となされるため、いずれにせよ認証機能は必要である。この認証機能は、注文と切り離して別途設けることもできるが、例えば、チャレンジを行ないそれに対する応答により認証する方法では、これらチャレンジおよび応答のやりとりが、實際上、ソフト実行器のユーザに対してかなりの負担を与えることになる。そこで、第3の実施例では、注文情報およびソフト実行器の認証を、注文と同時に効率的に行なうようにしている。そのため、第3の実施例では、注文情報および実行器の暗号キーに依存した注文認証情報を導入している。そして、この注文認証情報は、注文情報および実行器IDに加えられて、注文管理者302に伝えられる。なお、この注文認証情報は、上記第3の実施例ではハッシュ関数の出力であるため、例えば10進10桁程度で実現でき、電話で伝える場合にもそれほどユーザに負担を与えない。

【0097】注文管理者302は、上記注文認証情報を用いることにより、注文情報およびソフト実行器301の認証を行なっている。つまり、ソフト実行器301が注文作成部4で作成した注文情報と、注文管理者302に電話で伝えた注文情報とが異なっている場合には、注文管理者302の実行器認証部32において、NGが出力される。また、あるソフト実行器が自分の実行器IDと異なる情報を電話で伝えた場合にも、この実行器認証部32でチェックできる。

【0098】上記認証結果がOKの場合にのみ、注文管

理器302は、注文認証情報を注文を識別するための注文識別情報として用いる。そして、注文管理者302は、この注文に対応した実行権生成パスワードを生成する。この実行権生成パスワードは、ソフト実行器301に伝えられる。この実行権生成パスワードもハッシュ関数の出力であるため、10進10桁程度で実現できる。ソフト実行器301では、入力された実行権生成パスワードを、内部に蓄えている注文情報および注文認証情報(=注文識別情報)を用いて検査する。この検査を経た後、入力された実行権生成パスワードに対応した注文認証情報および注文情報を保持する特定のソフト実行器でのみ有効となる実行権が生成される。なお、ここでは、実行権生成パスワードの注文識別情報(=注文認証情報)の中に、実行器暗号キーの情報がすでに含まれているため、実行権生成パスワード自身に実行器暗号キーは含まれていない。

【0099】上記第3の実施例では、ソフト実行器301のユーザは、注文時に第1の実施例で伝達すべき情報に加えて、注文認証情報を注文管理者302に伝えている。このことにより、注文管理者302は、実行権生成パスワードを発行する前に、注文情報およびソフト実行器の認証を行うことができる。この注文認証情報は、10進10桁程度の値であり、ソフト実行器301のユーザに対してそれほど負担を与えない。また、第1の実施例と同じ回数のやりとりで認証を実現でき、ユーザ認証を別途設ける場合に比べて効率がよい。そして、この注文認証情報は、注文情報を識別する注文識別情報として、注文情報やそれに関係する情報を管理するのに用いられる。

【0100】(4) 第4の実施例

図4は、本発明の第4の実施例に係るソフトウェア保護システムの構成を示すブロック図である。この第4の実施例は、乱数を導入した第2の実施例に、効率のよい実行器認証機能を付加したものである。第4の実施例も第2の実施例と同様に、例えばソフトの実行権に実行回数などの条件が付加されている場合に有効となる。図4において、第4の実施例のソフトウェア保護システムは、ソフトウェアを実行するソフト実行器401と、ソフト実行器401に対してソフトウェアを配付する注文管理者402とを備えている。

【0101】ソフト実行器401は、前述の第1の実施例と同様の構成の、暗号化ソフトウェア格納部3と、注文作成部4と、実行器ID格納部5と、実行器暗号キー格納部6と、実行権生成部8と、実行権格納部9と、ソフト復号実行部10とを含む。さらに、ソフト実行器401は、注文情報および実行器暗号キーを結合してハッシュ演算を行う注文ハッシュ部40と、予め定められた構造や意味をもつ乱数を注文ごとに生成する乱数生成部41と、当該乱数および注文ハッシュ部40の出力の排他的論理和を演算する排他的論理和部42とを含む。

【0102】一方、注文管理者402は、図1の実施例と同様の構成の実行器暗号キー格納部11を含む。さらに、注文管理者402は、実行器暗号キー格納部11から獲得した実行器暗号キーおよびソフト実行器401から受け取った注文情報を結合してソフト実行器401と同じハッシュ演算を行う注文ハッシュ部44と、ソフト実行器401から受け取った注文番号および注文ハッシュ部44の出力の排他的論理和を演算する排他的論理和部45と、排他的論理和部45の出力を入力しソフト実行器401の正当性を認証する実行器認証部46と、実行器認証部46の認証結果がOKのときのみ起動して実行権生成パスワードを発生する実行権生成パスワード生成部47とを含む。

【0103】次に、図4に示す第4の実施例のソフトウェア保護システムの動作を説明する。まず、ソフト実行器401において、注文作成部4が注文情報を作成すると、注文ハッシュ部40は、第1の実施例で用いたのと同様のハッシュ関数に、当該注文情報と実行器暗号キー格納部6から得たソフト実行器の暗号キーとを結合したデータを入力する。また、乱数生成部41は、予め定められた構造や意味を持った乱数データを生成する。一例として、本実施例では、乱数データの最上位から3ビットを“0”とし、最下位から3ビットを“1”としている。その他のビットは乱数を格納する。この例の他、注文管理者402とソフト実行器401との間で予め取り決めておいて、その乱数データのある部分に、実行器IDの一部のデータを対応させてもよい。次に、排他的論理和部42は、注文ハッシュ部40の出力と、乱数生成部41で生成された乱数データとの排他的論理和を演算する。ソフト実行器401のユーザは、この排他的論理和部42の演算結果を注文認証情報として、注文情報および実行器IDと共に、注文管理者402のオペレータに伝える。なお、上記乱数データのビット数を、注文ハッシュ部40の出力と同程度にとれば、注文認証情報は10進10桁程度となり、ソフト実行器401のユーザや注文管理者402のオペレータにそれほど大きな負担を与えない。

【0104】注文管理者402では、実行器認証部46が実行器暗号キー格納部11から該当するソフト実行器の暗号キーを獲得し、これと注文情報とを結合して注文ハッシュ部44に入力する。そして、排他的論理和部45は、注文ハッシュ部44の出力と、ソフト実行器401から受け取った注文認証情報との排他的論理和を演算する。注文情報と実行器IDとが正しければ、注文ハッシュ部44の出力は、注文を行ったソフト実行器401内の注文ハッシュ部40の出力と同じになる。また、排他的論理和部45の出力は、注文を行ったソフト実行器401の乱数生成部41で生成した乱数データと同じになる。つまり、今の場合、乱数データの最上位から3ビットは“0”、最下位から3ビットは“1”となる。実

行器認証部46は、排他的論理和部45の出力が、予め定められている乱数の構造や意味を持っているか否かを確認し、OKであれば、注文情報と実行器とが正しいと認証する。そして、この認証結果がOKの場合にのみ、実行権生成パスワード生成部47は、実行権生成パスワードを生成する。これ以降の動作は、第3の実施例と同じである。すなわち、注文管理者402で生成された実行権生成パスワードが、電話等によってソフト実行器401に伝えられ、ソフト実行器側でそれを検査して、OKの場合に対応するソフトの実行権が作成される。

【0105】上記第4の実施例では、第2の実施例と同様に、乱数を導入して回数制限付きの実行権生成に対応できる。さらに、第4の実施例は、第3の実施例と同様に、ユーザ認証機能を備えている。このように第4の実施例は、第2および第3の実施例の機能を併せ持つにもかかわらず、ソフト実行器401と注文管理者402間でやり取りするデータ量、特にランダムな注文番号と実行権生成パスワードの桁数が増加していない。

【0106】(5) 第5の実施例

図5は、本発明の第5の実施例に係るソフトウェア保護システムの構成を示すブロック図である。この第5の実施例では、各ソフト実行器内に不活性状態の実行権生成モジュールが備えられており、注文管理者から与えられる実行権生成パスワードによりその実行権生成モジュールを活性化させて、そのモジュールで所定の実行権を生成することを特徴としている。なお、以下の説明では、実行権生成モジュールが2つの部分に分けられ、その一方である実行権生成モジュールAがソフト実行器に蓄えられ、もう一方の実行権生成モジュールBは、暗号化されてソフト実行器が注文管理者から受け取る。この実行権生成モジュールAおよびBは、両者が揃って初めて所定の実行権が生成できる。従って、ソフト実行器内に蓄えられた実行権生成モジュールAだけでは不活性な状態であり、実行権を生成できない。

【0107】図5において、第5の実施例のソフトウェア保護システムは、ソフトウェアを実行するソフト実行器501と、ソフト実行器501に対してソフトウェアを配付する注文管理者502とを備えている。

【0108】ソフト実行器501は、前述の第1の実施例と同様の構成の、暗号化ソフトウェア格納部3と、注文作成部4と、実行器ID格納部5と、実行器暗号キー格納部6と、実行権生成部8と、実行権格納部9と、ソフト復号実行部10とを含む。さらに、ソフト実行器501は、注文情報および実行器暗号キーを結合してハッシュ演算を施す注文ハッシュ部50と、実行権生成モジュールAを格納する格納部51と、注文管理者502から受け取った実行権生成パスワードを逆変換する逆変換部52と、逆変換部52の逆変換結果を格納する格納部53とを含む。なお、注文情報と実行器IDとが正しいければ、この格納部53には、実行権生成モジュールBが

格納されることになる。

【0109】一方、注文管理者502は、図1の実施例と同様の構成の実行器暗号キー格納部11を含む。さらに、注文管理者502は、ソフト実行器501から受け取った注文情報と、実行器暗号キー格納部11から得た該当の実行器の暗号キーとを結合してハッシュ演算を施す注文ハッシュ部54と、実行権生成モジュールBを格納する格納部55と、モジュールBを注文ハッシュ部54の出力を用いて変換し、実行権生成パスワードを生成する実行権生成パスワード生成部56とを含む。

【0110】次に、図5に示す第5の実施例のソフトウェア保護システムの動作を説明する。まず、ソフト実行器501のユーザは、ソフト実行器501を操作することにより、実行を希望するソフトウェアの注文を指定する。応じて、注文作成部4は、対応する注文情報を作成する。そして、当該ユーザは、注文情報とソフト実行器501に固有の実行器IDとを、電話等によって注文管理者502に通知する。

【0111】ソフト実行器501からの通知を受け取ると、注文管理者502では、実行器IDに基づいて実行器暗号キー格納部11を検索し、該当するソフト実行器の暗号キーを獲得する。次に、注文ハッシュ部54は、ソフト実行器501から受け取った注文情報と、実行器暗号キー格納部11から得た実行器暗号キーとを結合して、第1の実施例で用いたのと同様のハッシュ関数に入力する。実行権生成パスワード生成部56は、注文ハッシュ部54から出力されるハッシュ値を鍵として、実行権生成モジュールBを変換し、実行権生成パスワードを発行する。注文管理者502のオペレータは、この実行権生成パスワードを、電話等によってソフト実行器501のユーザに伝える。

【0112】次に、電話で実行権生成パスワードを受け取ったソフト実行器501のユーザは、ソフト実行器501の逆変換部52に、このパスワードを入力する。注文ハッシュ部50には、注文管理者502で用いたのと同じハッシュ関数が備えられている。そして、注文ハッシュ部50は、注文作成部4内に蓄積された注文情報と、実行器暗号キー格納部6から得た実行器暗号キーとを結合し、この結合データに対してハッシュ演算を施す。逆変換部52は、注文ハッシュ部50から出力されるハッシュ値を鍵として、注文管理者502から受け取った実行権生成パスワードを逆変換する。もし、注文情報と実行器暗号キーとが正当であるならば、注文ハッシュ部50から出力されるハッシュ値は、注文管理者502におけるハッシュ値（注文ハッシュ部54の出力）と同じになる。従って、逆変換部52の出力は、実行権生成モジュールBとなる。次に、実行権生成部8は、予め格納部51に格納されている実行権生成モジュールAと、獲得したモジュールBとを用いて、注文情報に対応した実行権を生成する。実行権生成部8によって実行権

が生成された後は、実行権生成モジュールに従って、モジュールBが削除される。

【0113】上記第5の実施例では、ソフト実行器側に予め格納されている実行権生成モジュールAは、不活性な状態であり、情報量的にこれだけでは動作できない。これを活性化するためのモジュールBは、注文管理者502において、ソフト実行器の注文情報と暗号キーとに依存した形に変換され、実行権生成パスワードとして送付される。前述した第1～第4の実施例においては、実行権生成モジュールは、完全な形でソフト実行器内に存在した。ただし、これを起動するためには、実行権生成パスワードという許可が必要であった。これに対して第5の実施例の構成では、実行権生成モジュールは、情報量的に不足した状態でソフト実行器内に存在する。このため、第5の実施例は、第1～第4の実施例に比べて、より実行権の不正生成に対する安全性が高い。なお、実行権生成モジュールAおよびBにより注文の実行権を生成した後は、元の状態に戻すために、モジュールBが削除される。

【0114】第5の実施例において、実行権生成モジュールBは、注文情報と実行器の秘密キーとに依存して変換されるため、注文が異なったり実行器が異なる場合には、ソフト実行器のユーザは、正規のモジュールBを手に入れることができない。ただし、第5の実施例では、実行権生成パスワードの中にモジュールBが情報として含まれるため、許可情報だけが含まれていた第1～第4の実施例に比べて、多少実行権生成パスワードの桁数を増やす必要がある。なお、この第5の実施例における実行権生成パスワードのデータ量は、注文するソフトの個数に依存しない。

【0115】なお、第5の実施例では、ソフト実行器501は、注文管理者502から受け取った実行権生成パスワードをチェックすることなしに、逆変換後すぐにこれをモジュールBとして用いている。従って、もし注文情報や実行器暗号キーが、入力した実行権生成パスワードに適合しない場合には、実行権生成モジュールが正しく動作しない。ところで、例えばモジュールBがある特定の構造や意味を持っているとし、ソフト実行器501において逆変換後、これを使用する前に構造や意味をチェックすることも考えられる。例えば、モジュールBがある命令コードの場合には、そのコードには特定の構造があるはずである。従って、モジュールBがこの特定の構造を有しているか否かをチェックすることにより、実行権生成モジュールの実行前にエラーを処理することができる。

【0116】なお、以上述べた第2および4の実施例において用いられる乱数は、注文ごとに変化する値であれば良く、カウンタなどを用いて発生してもよい。また、乱数に代えて、タイムスタンプを用いてもよい。タイムスタンプは、ソフト実行器と注文管理者とで共通に発生

可能（時計回路によって発生可能）であるため、ソフト実行器から注文管理者に対して変化値を送付する必要がなくなる。

【0117】また、第3および4の実施例において、実行権生成パスワードを生成する際、注文情報および注文識別情報に加えて、実行器暗号キーにも依存させるようにしてもよい。ソフト実行器にとって、自分の実行器暗号キーを知っているのは、自分以外には注文管理者だけなので、この実行権生成パスワードを作ることができるのは、注文管理者だけである。そのため、実行権生成パスワードを注文管理者の署名情報として考えることができ、トラブル等が生じたときに、この情報を第3者に証拠として提出することができる。また、このような秘密鍵暗号を用いた署名方法に代えて、公開鍵暗号を用いた署名方法を用いてもよい。この場合、注文管理者は秘密の情報を保持し、実行権生成パスワードの生成の際にこの秘密の情報をを用いる。一方、ソフト実行器側では、注文管理者の秘密情報に対応する公開情報を保持しており、これを用いて実行権生成パスワードの正当性を確認する。注文管理者の秘密情報を用いた場合にのみ、実行権生成パスワードの正当性が確認できるため、この実行権生成パスワードを注文管理者の署名として考えることができる。

【0118】また、第1～4の実施例においては、元のソフトウェアSoftAと、対応する認証子authAとを結合し、この結合されたデータをシステム共通の鍵Sで暗号化することにより、ソフトウェアの暗号化を行うようにしていたが、また、この認証子を実行器暗号キーSxで暗号化したデータを実行権として用いるようにしていたが、本発明はこのような方法には限定されない。例えば、認証子と実行器IDとを結合し、この結合されたデータをシステム共通の鍵で暗号化したものを、実行権として用いてもよい。実行器Xの実行権を、式で記述すると、

$$E(S, authA \parallel IDx)$$

となる。この場合、実行権は、実行権と暗号化ソフトウェアとをシステム共通の鍵で復号し、両者から得られる認証子が一致しており、かつ、実行権の復号結果から実行器IDを獲得して、これが実行器内の実行器IDと一致している場合に認証され、ソフトウェアを実行する。いずれにせよ、暗号化ソフトウェアから実行器がすでに保持しているデータを用いて、実行権が生成できる暗号方式ならば、本発明を適用できる。

【0119】また、上記第1～4の実施例においては、電話の1通話で注文を行ない、それに対する実行権生成パスワードを受け取るとしていたが、これが別の通話であってもよい。つまり、まずソフト実行器から注文管理者に対して注文を行ない、この注文を一旦セーブする。次に、注文管理者がソフト実行器に電話をかけてきて、注文番号で照合を行ない実行権生成パスワードの通知を

行なう。このことにより、ソフト実行器のユーザにとり、電話をかけている時間が短くなり、料金的にも都合がよい。また、ソフト実行器のユーザは、注文管理者のユーザから実行権生成パスワードが通知されるまで、別のソフト実行等の作業を行なうことができる。一方、注文管理者にとっても、コールバックを行なうことによってユーザ認証が行なえる。

【0120】また、上記第1～4の実施例において、ソフト実行器が実行権生成パスワードを検査して実行権を生成する場合には、予めセーブしておいた注文情報をロードして用いる。ただし、注文情報を再ロードし、パスワードを再度使われないようにするために、最初に実行権を生成した後、セーブしておいた注文情報を消去することが必要となる。

【0121】また、上記各実施例において、ソフト実行器と注文管理者とのやり取りの履歴を、双方で残しておくことにより、注文に関するトラブル時の参考資料にすることが考えられる。また、ソフト実行器側には、例えば今までどんなソフトを購入したかを残しておくことにより、注文ガイダンスのユーザインタフェースを良好なものにすることができる。

【0122】また、上記各実施例において、ソフト実行器から注文管理者への注文情報としては、例えばソフトAとソフトBという具合にソフトのIDを用いることにしていたが、この部分をコード化して、対応するソフト名とコードの対応表を用いることにより、より通信情報量の削減を行なうことも考えられる。つまり、ソフトAとソフトBならばコード2番、ソフトBだけならばコード10番といった具合にである。

【0123】また、上記各実施例では、ソフト実行器と注文管理者との間の情報のやりとりは、人間が電話を介して行うものとして説明したが、両者に送受信器を設け、通信路を介して情報の授受を行うようにしてもよい。

【0124】

【発明の効果】請求項1の発明によれば、ソフト実行器は、注文に対応した正規の実行権生成パスワードを入力することによってのみソフトの実行権を生成する。そして、注文管理者からソフト実行器には、実行権生成パスワード、すなわちソフト実行器が注文情報に対応したソフト実行権を生成するための許可を与える情報だけが送付されるため、安全性を劣化させることなく、送付すべき情報量を大幅に削減することができる。また、実行権生成パスワードの情報量は、注文するソフトの個数に依存せずに、いつも少ない情報量で実現できる。すなわち、ソフト実行器が複数のソフトウェアの注文を1回に行なっても、その情報量は1つのソフトウェアを注文した場合の情報量と同じになる。この情報量の削減により、特に電話でパスワードを送受信する場合、ユーザインタフェース的に都合がよい。また、注文管理者にとつ

ては伝え間違いが少なくなる。またソフト実行器のユーザにとっては、聞き間違いおよびこれを実行器に入力する際に間違いが少なくなる。また、電話をつないでいる時間も削減できるため、料金的にもメリットがある。

【0125】請求項2, 5, 8, 10および15の発明によれば、データ圧縮関数を用いて実行権生成パスワードを生成するようにしているので、その情報量をより一層削減することができる。

【0126】請求項3の発明によれば、ソフト実行器には、デジタル署名された実行権生成パスワードが送付されるので、ソフト実行器側で実行権生成パスワードを偽造することがほとんど不可能となり、極めて安全性の高いソフトウェア保護システムを実現することができる。

【0127】請求項4の発明によれば、注文情報を作成する毎に変化する変化値を導入し、注文管理者からの実行権生成パスワードは、対応する変化値を保持するソフト実行器でのみ有効となるように構成されているため、異なる時期に同じ注文を作成し、以前の実行権生成パスワードを入力しても、後の注文時に生成される変化値は、以前の注文時に生成される変化値とは異なっており、入力したパスワードが有効にはならない。これによって、例えば実行の回数が制限されているような実行権が生成される場合、実行権生成パスワードを繰り返して用いるという不正を防止することができる。

【0128】請求項6の発明によれば、異なる時期になされた注文を峻別するために、ソフト実行器と注文管理者とに共通に保持されているタイムスタンプを用いているので、ソフト実行器から注文管理者に峻別のための情報を送付する必要がない。

【0129】請求項7の発明によれば、ソフト実行器側で注文情報および実行器秘密情報に依存した注文認証情報を生成し、注文管理者に送付するようにしているので、注文管理者側で実行権生成パスワードの発行に先だって、注文情報とソフト実行器の認証を行うことができる。注文情報とソフト実行器の認証は、実際上は課金に関係するため、必要不可欠な機能である。本発明では、注文と同時にソフト実行器から少ない桁数の注文認証情報を通知することによって、効率的にこの機能を実現している。また、注文認証情報を、注文を識別する注文識別情報としても用いることができる。

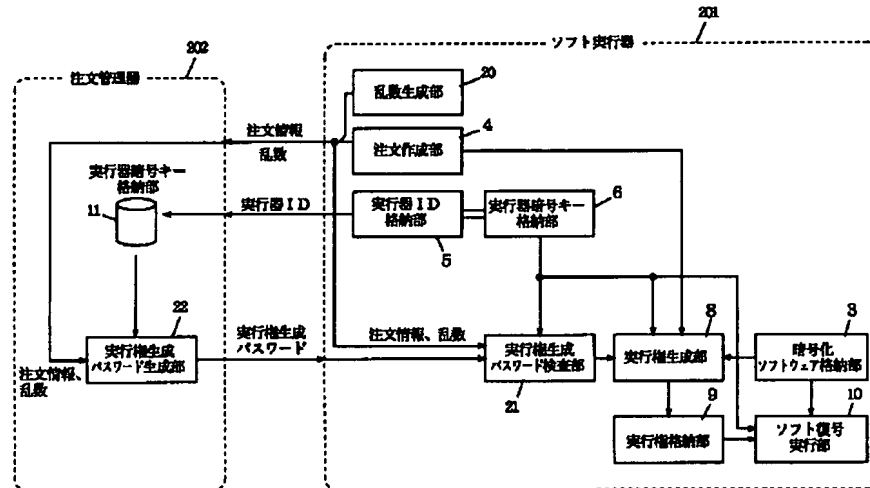
【0130】請求項9の発明によれば、ソフト実行器と注文管理者間でやりとりされる情報の量を増やすことなく、回数制限機能およびユーザ認証機能の両方を実現することができる。

【0131】請求項12の発明によれば、注文管理者からソフト実行器には、デジタル署名された実行権生成パスワードが送付されるので、ソフト実行器側で実行権生成パスワードを偽造することがほとんど不可能となり、極めて安全性の高いソフトウェア保護システムを実現することができる。

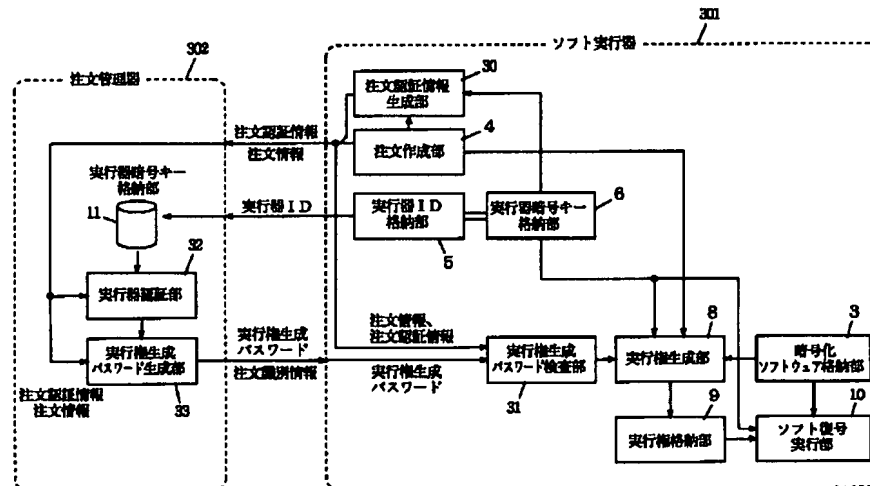
【図2】本発明の第2の実施例に係るソフトウェア保護システムの構成を示すブロック図である。

56…実行権生成パスワード生成部

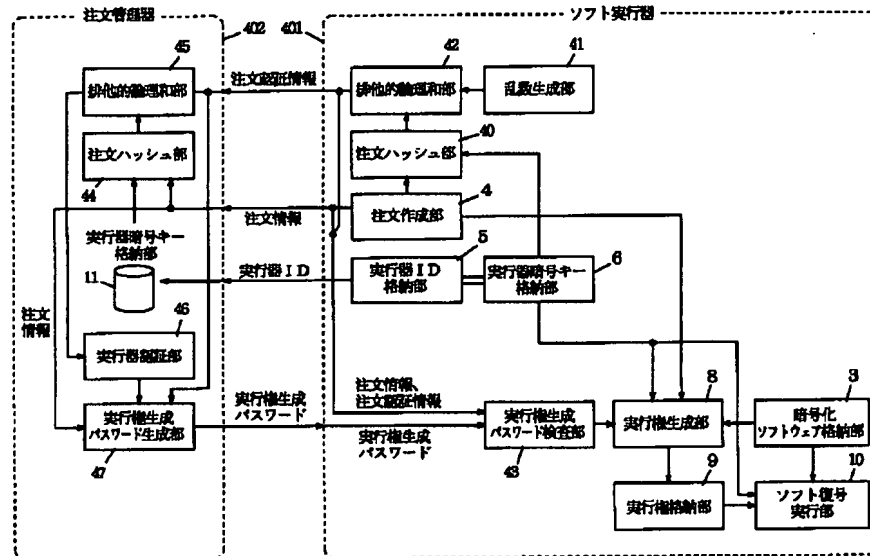
【図2】



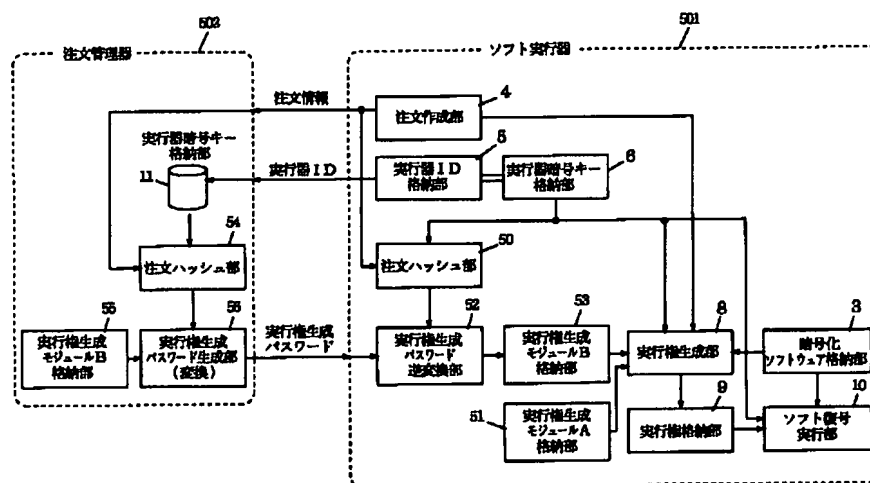
【図3】



【図4】



【図5】



【図６】

